Prepress version

Ackerman, Mark S. and Christine Halverson. "Sharing Expertise: The Next Step for Knowledge Management." Wulf, Volker and Marlene Huysman (eds.), Social Capital and Information Technology. MIT Press, 2003

Sharing Expertise: The Next Step for Knowledge Management

Mark S. Ackerman
School of Information and Department of Electrical Engineering and Computer Science
University of Michigan
MarkAckerman@umich.edu

Christine Halverson
Social Computing Group
IBM T.J. Watson Research Center
krys@us.ibm.com

1 Introduction

There are numerous ways to handle knowledge within organizations. Indeed, knowledge management has been a flourishing commercial area for almost ten years, and one can point to many precursors within organizations as well. Knowledge management – regardless of its title or position in history – has always been an important, though not necessarily frequent, aspect of organizational life. It would be difficult to imagine a modern corporation that did not occasionally reflect and improve its methods of handling communications, data, and information – or try to learn from its experience. In this chapter, we move from the metaphor of *knowledge management* to a new metaphor, *expertise sharing*, which promotes focusing on the inherently collaborative and social nature of the problem.

In our view, "knowledge management" subsumes a number of differing strategies. What all of these strategies share - as do many information access strategies - are interactions with or foundations in the social setting of an organization or institution. This is made explicit in descriptions of social capital. While social capital itself can be many things, we consider here its use within an organizational knowledge management definition [Cohen and Prusak 2001, Lesser, Fontaine, and Slusher 2000, Stewart 2001, Wenger 1998]. (See the excellent review of the definitions and connotations of social capital in Syrjänen and Kuutti [this volume].)

Social capital, as described by Huysman ([Huysman 2003], following [Nahapiet and Ghoshal 1998]), has three aspects: a structural dimension, a cognitive dimension, and a relational dimension. As Lesser and Prusak [2000] point out, "…the structural dimension of social capital reflects the need for individuals to reach out to others within an organization to seek out resources that they may not have at their disposal. (p. 126)" It consists of network ties and their configuration and organization. The cognitive dimension includes shared language and common narratives; hence, it includes the social-cognitive aspects of an organization. The relational dimension includes trust, norms, obligations, and a shared identification. As Lesser and Prusak note, "…social capital is developed and fostered when individuals believe that their actions will be appropriately reciprocated and that individuals will meet their expected obligations. (p. 127)"

Our contention here is that the structural, shared cognitive, and relational dimensions are the social aspects that must be incorporated into next-generation knowledge management systems. The structural, shared cognitive, and relational dimensions of an organization allow knowledge and expertise to be shared. Expertise sharing, then, requires a deep consideration of how social capital must unfold throughout knowledge management. Only in considering these components of an organization can anything approaching knowledge management or organizational memory be achieved in practice.

Accordingly, we will first show how the four standard mechanisms for sharing expertise and managing knowledge suffer from various collaborative and social issues. Underlying these issues is one of the intellectual challenges facing computer-supported cooperative work (CSCW or groupware) as a field. The gap between what we know we have to do socially and what computer science as a field knows how to do technically (what we call here the social-technical gap) has led the two of us to reflect on potential systems designs in order to ameliorate this social-technical gap. Accordingly, the last half of the chapter is a review of our research that has evolved into a more organizationally attentive direction. Our description of these research systems focuses on their incorporation of, or augmentation to, the structural and relational aspects of social capital. (We will largely leave the cognitive aspects aside, as we believe that much of knowledge management assumes these. In addition, we will use the terms "social-structural" and "social-relational" to differentiate them from the technology terms "structural" and "relational".) We conclude with some potential future research directions.

2 Current technical possibilities for knowledge management

Broadly speaking, there are four technical directions that knowledge management or expertise sharing systems take at present. They align along a dimension that ranges from "objectified" knowledge decontextualized and separated from individuals and placed into repositories to "embedded" and "community" knowledge found in groups of individuals. These should not be viewed as a progression; each has its place. One of the purposes of this chapter is to understand the tradeoffs involved in handling expertise or knowledge in each manner.

The first technical possibility is a *repository*. Typically, this consists of a data store of "knowledge" fragments. These are similar to, or sometimes the same as, corporate databases. An *expertise locator* is a recommendation engine or "yellow pages" directory that helps people find other people with the expertise that is required for some activity. A *computer-mediated place* (e-community, *knowledge community*, or computer-mediated communication system) is a virtual space where people with questions or answers can gather. Finally, there is hope that one can collect people into *ad-hoc groupings*, flexible arrangements of an organization's social network in order to solve specific, time-limited problems.

We will cover each in turn.

2.1 Repositories

The original vision for a knowledge repository was relatively simple: A company should be able to remember what it has done previously. The idea, then, was to put that previous experience, or knowledge, in an information-base of some sort. These efforts ranged from data warehouse initiatives to new forms of text databases such as Lotus Notes.

Perhaps because repositories were the first technical augmentation in this knowledge wave, the issues are perhaps best understood. This vision of knowledge repositories had four major weaknesses. First, proponents felt that they could construct one information-base for an entire company. This was quickly discarded, as the political realities and technical issues involved became apparent. Second, there was an assumption that all knowledge could be removed from individuals and placed into an information-base. This was discarded, although more slowly, as proponents began to discover which information could be decontextualized appropriately or even made explicit. Third, it was assumed that people would share their knowledge spontaneously. Finally it was assumed that people would naturally understand what others had put into the information-base.

These problems result largely from not understanding the social and organizational dimensions, both social-relational and social-structural, of repositories. In an organization, information is not value-free. Nor is sharing free – it carries psychological costs, and the rewards may be unclear. If others plan to use the information, then it takes time and effort to properly write up the information. That is, the writer must go through the effort of properly decontexualizing the information, and then the reader must go through the effort of recontextualizing it to her own needs. Of course, storing knowledge "objects" is not the same as understanding the social processes that surround knowledge acquisition, dissemination, and understanding.

2.2 Expertise locators

After people began to see the limitations of knowledge repositories, they began to explore how people might also provide information and knowledge to others. Accordingly, we have viewed expertise location as either finding the right person to answer the right question or finding a person to complement a team appropriately.

Many approaches to expertise location or expertise finding have been proposed. (See [Ackerman, Pipek, and Wulf 2002] for a summary of the various approaches and selected systems.) The major difficulty with these approaches is keeping the finder engines stocked with up-to-date information about people. How to typify people, skills, and expertise is an open question. The dimensions of description are unclear, especially for social-relational issues, and it is difficult to keep abreast of dynamically changing situations. Even though a company might have undertaken a skills inventory, new requirements arise and need to be included. For example, a company may have learned that Joe knows the C programming language well, but now needs to know who has learned the Perl and C# languages. We might also want to know how easy it is to interact with Joe or others (social-relational), or who else has overlapping skills if Joe is not available (social-structural). For expertise locators, then, not only must the engine's recommendations be accurate in its data, the data must also be correct, timely, and organizationally relevant [Ackerman et al. 2002].

2.3 Computer-mediated places

One way to find expertise currently residing in people is to find the people themselves. An alternative, however, is to have the people come to the problems. In other words, one would like to have an online place where people can go to have questions answered. This requires an electronic community, also called online community, virtual community, or community of practice, where people with expertise congregate and are able to help relative newcomers (and experts) along [Wenger 1998].

The major issue with this vision is that it assumes others will want to join in. Strong social ties, those existing between people with a history of social interactions, do lead to people spending time and energy. This could be fostered by having people spend time together, and since in a large, multinational corporation this might be impossible to do face-to-face, it would have to be done through online, computer-mediated places. However, interesting people with expertise do not necessarily have the time to hang out together waiting for questions. Furthermore, not everyone is friendly or sociable. For example, Allen [1977] found that his gatekeepers, people who knew other people in an organization, were far more likely to be sociable (and productive) than were others in an organization.

2.4 Ad-hoc groups

As we understand how to reconstitute the social network of an organization through new communication mechanisms [Sproull and Kiesler 1991], there arise possibilities of creating ad-hoc groupings or subnetworks. These groupings can come together quickly, actively work on a problem, and then disband after they have finished. Organizations have had crisis teams or tiger teams, but new technologies have offered the vision of supporting geographically distributed and extremely short-duration teams. These teams are the virtual equivalent of the recent interest in the extreme programming teams [Teasley et al. 2000].

This work is being attempted currently, but two major issues with this vision have already surfaced. First, it is difficult to make up a team that can work together. Doing so entirely virtually is even more difficult. Ameliorating this difficulty, however, could be strong organizational cultures or earlier face-to-face work. The second issue is the same as for expertise locators – the great difficulty of finding the data to know who does what well. As well, those putting together teams need to understand potential team members' work styles and other influences salient to accomplishing the work.

The above sections have attempted to indicate the current technical possibilities and show how their potential use is deeply influenced by organizational and social issues. The following section argues that this influence is not a product of the problem per se. Sharing expertise requires an understanding of the social and organizational because it is deeply collaborative, and collaborative systems suffer from a so-called "technical-social gap". The following section summarizes this gap, and shows why it cannot be avoid. It should be noted that many others have also argued for the existence of this gap; this section merely serves to show how sharing expertise is influenced heavily by this gap. After providing a brief overview of the gap, Section 4 will show some of our attempts to move around this gap in order to share expertise within organizations.

3 The social-technical gap

Each of these technical mechanisms for sharing expertise has problems and issues, as the last section pointed out. These result from deep, underlying social issues. Similar to much of computer-supported cooperative work (or groupware), a research area that studies how groups or organizations use technology, knowledge management both gains and suffers in the interaction between technology and social phenomena. By concentrating merely on surface issues, knowledge management efforts have fallen below expectations. We shall argue below for a theoretical reason for this, and we shall show why seriously including social-structural and social-relational considerations (i.e., social capital) into technology must be rewarding but inherently difficult.

There is a set of well-known findings that are almost assumptions within CSCW. A summary of these findings can be found in [Ackerman 2000] and [Ackerman 2001]. In a large sense, these are social findings that we know we need to deal with when building these systems.

This section discusses three of these social issues. The selection of these three is somewhat arbitrary, in that many others could have been chosen. The three are:

- Impression management
- Negotiating norms
- Incentive structures

As will be discussed further below, each of these social issues is a well-known problem in constructing or using CSCW systems, but solutions remain illusive.

3.1 Impression management

Humans are very good at social interaction. People have very nuanced behavior concerning how and with whom they wish to share information. Goffman [1961] noted that people present "faces" to one another: We present different information to fellow researchers, students, spouses, and even relatives. What we tell our mothers is not what we tell our best friends about ourselves. What we tell spouses about job prospects and career goals is not necessarily what we tell our managers. Goffman argued that a critical component of one's social psychology is the ability to do what he called "impression management". Indeed, people in face-to-face interactions find it very disconcerting to lose control of what they consider private information. The first author remembers a colleague becoming almost violently angry because word of an award had leaked out before he could mention it – imagine what it would be like for truly private information. Goffman was fascinated by spies and thieves, but everyone does impression management and wishes to maintain control.

Computational systems, such as those for sharing expertise, are notoriously poor at helping people do impression management. Access control systems often have very simple models. Our social activity is fluid and nuanced [Garfinkel 1967, Strauss 1993, Suchman 1987], and we do impression management almost without thinking about it. Indeed, in face-to-face interaction, we would find it strange to consciously consider each person, write down their category on a slip of paper, and then continue to communicate the details of our lives. Any access or security control system, on the other hand, tends to get in the way of impression management and the underlying social interaction, rather than to foster them. We believe there is an inherent tension here – one either has the nuance of control or the fluidity of interaction. Efforts to obtain both will necessitate the automatic inference of context; this is obviously very difficult. As a result of this computational gap, sometimes it is easier and better to augment technical mechanisms with social mechanisms to control, regulate, or encourage social behavior [Sproull and Kiesler 1991].

3.2 Norms of use are negotiated.

People create norms to govern how they act in social settings. Norms have been considered as formal as rules or as informal as guidelines; for our purposes, here we can use the term somewhat loosely to cover the entire spectrum [Feldman 1984]. A collaborative system, such as a knowledge management system or an expertise location system, is no different than any other social setting: People still create and use norms to govern their social interactions. As with face-to-face or other social settings, the norms for using a collaborative system are often actively negotiated among users. That is, unless use is mandated and strictly controlled by a governing hierarchy (and perhaps even in those situations), the users themselves will have to work out “the rules of conduct”. These norms cannot be merely looked up in a rulebook of some sort – exceptions and new situations occur, people with new needs or new abilities arrive, and formal rules are often too inflexible to get the actual work accomplished.

Accordingly, any norms of use are also subject to re-negotiation [Strauss 1991]. The people who use a system (or inhabit a social setting, however briefly) constantly interpret and re-interpret the norms of behavior, shaping to the current inhabitants and needs. Accordingly, collaborative systems often require some secondary mechanism or communication back-channel.

3.3 Incentives are critical

Grudin [1989] framed what is sometimes called the Grudin paradox: What may be in the managers’ best interests may not be in the ordinary users’ interests. In his analysis of group calendar systems, he noted that management would like to have employees’ schedules so they can be examined and managed. However, it is not in the interest of the employees to have their schedules open, if they achieve no other benefit from group calendar use. Grudin pointed out that the incentives for collaborative activities must be symmetrical; that is, there must be incentives and rewards for all users.

With expertise sharing systems, there must be incentives for experts as well as the other users [Orlikowski 1992]. For example, if users can ask questions of experts through a system, then clearly there is a benefit for the users. On the other hand, experts merely gain more overloaded inboxes and interruptions. Accordingly, the organizational reward system, culture, or work assignment needs to be readjusted to provide a benefit to the experts as well.

A related problem is that the use of collaborative systems is often tedious. Additional data may need to be entered. For example, users may need to enter access control information to resolve who can read highly proprietary information. Many CSCW researchers try to use available data to reduce the cost of sharing and collaborative work

3.4 The gap

The above three sets of social findings that have been known, in one form or another, for many years. However, we do not know how to construct systems that meet these findings [Ackerman 2000]. Unfortunately there is a gap between what we know how to do technically and what we know we have to do socially. For example, we cannot solve the impression management problem by adding more rules to a system; rules are brittle and require user intervention at inopportune times.

The gap, then, is between social requirements and our technical capabilities. This is not a new statement – it is merely a restatement of the difference between “technically working” and “organizationally workable”. Given the current state of the art technically, there is an inherent tension between technically feasible systems and organizationally feasible systems. The history of CSCW demonstrates this tension, and CSCW is to be lauded for its understanding and interest in this tension.

As mentioned, many other researchers have argued for the existence of this gap. Our interest has been to argue that without new forms of technology, the gap is permanent [Ackerman 2001]. Our second interest has been to consider how to edge around the gap. While the gap results from an inherent tension, there are clearly things that can be done to both acknowledge and ameliorate the social requirements. So, what should one do? The rest of this chapter describes our efforts to grapple with this question for systems that share expertise.

4 Some systems and possibilities

This section describes what we see as a combined organizational-technical or social-technical approach for expertise sharing. Through the development of a set of systems and associated social studies we have examined a number of possibilities for augmenting the location and sharing of expertise. These explorations incorporate both an understanding of the organizational or social realities as well as the technical possibilities. Below, we discuss three areas of research work, all of which attempt to find interesting points in the combination organizational-technical design space:

- Tying together repositories with networks
- Self-feeding expertise locators
- Lightweight social spaces

We discuss each in turn.

4.1 Combining repositories and networks

In a series of studies, we examined combining information repositories with social networks. Our interest was in the iterative construction of information over time. In these systems, the user asks a question, and some expert answers. The result, over time, is a resulting information store.

In the original Answer Garden system [Ackerman 1993, Ackerman 1994, Ackerman 1996, Ackerman 1998, Ackerman and Malone 1990], there are a set of commonly-asked questions for some topic, a way to seek the information if the answer is not in the information database, and most importantly a way to grow and correct the database.

In Answer Garden, a user comes to the system to find an answer to some question. She can browse through the information database by clicking through a set of diagnostic questions, browsing an outline or tree view, or through an information retrieval engine. Figure 1 shows the interface for the original version in the X Window System; Figure 2 shows a part of a Web version. (There exist alternative interfaces and Answer Garden implementations from third parties.)

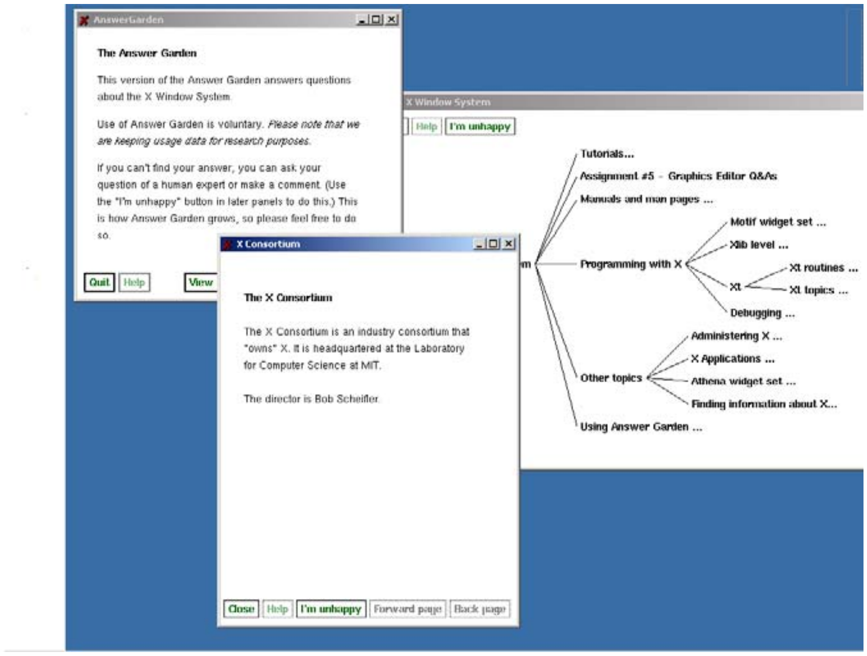


Figure 1: Answer Garden, the X Window System version.

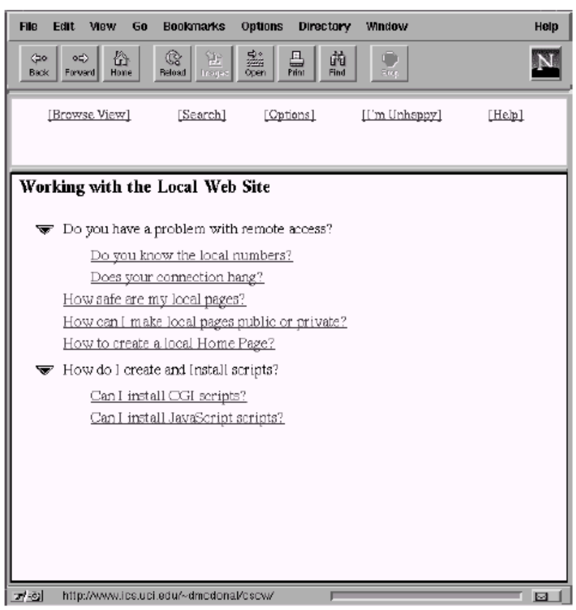


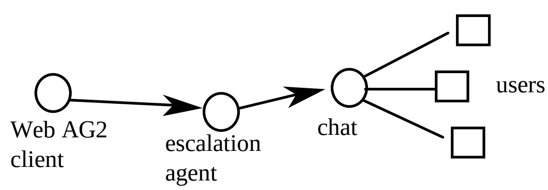
Figure 2: Answer Garden, the Web version.

If the user finds his answer, then he is finished. If the user does not find an answer, is confused with the answer or the navigation, or finds the answer incomplete, he can pop up a mailer (through the "I'm unhappy" button or link). He asks his question, and the system routes the question to an appropriate human expert. The expert answers. If the question and answer are common enough, the expert can insert the question-answer pair into the information store. This gave the system its name, since the system grows over time where and when users demand extra information. Users get answers, experts get rid of their commonly asked questions, and the organization as a whole gets a collaborative memory or knowledge repository.

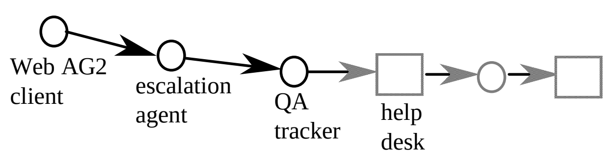
Importantly, the Answer Garden system does not separate getting information from information repositories and from people. In fact, users found this completely natural. When they could not find the answer in the information database, they were very satisfied to have the ability to use the organization's social network.

Answer Garden 2 [Ackerman and McDonald 1996] added the ability to route questions to many forms of computer-mediated communication. Escalation agents, consisting of rules in our sample implementation, could "gracefully escalate" the question to a chat group (or now, instant messenger list) of people nearby, then to a bulletin board, then perhaps to a help desk or consultant, and finally to an "expert" (Figures 3a and 3b). Answer Garden 2 corrected several issues in the original version. First, it eliminated the clear separation between experts and users. After all, many people have some level of expertise, and the true "expert" is a very scarce and expensive resource in any organization. Second, the people nearby the user are the most likely to understand the user's context. Since they know the user, they can also make best judgments about how to present the answer. Of course, when no knowledgeable people are nearby, there is an organizational dysfunctionality, and Answer Garden 2 still provides for getting an answer.

These two systems augmented the repository model by not only making access to commonly needed information easier, but also making access to people with the requisite knowledge easier. They tied people into the information system, while providing incentives for everyone involved. Their design deliberately rides the gap between social and technical in an attempt to bridge the limitations of both approaches.



(a) The user's first attempt to get an answer goes to a chat channel.



(b) The user's jth attempt to get an answer gets escalated to a help desk.

Figure 3: Answer Garden 2 functionality. Two possible escalations for a question.

Our approach to combining repositories with social networks has included both technology development and social studies. Some of our social studies are focused on technical considerations: Attempts to understand how the systems can be used (e.g., the adoption studies in [Ackerman 1994], [Ackerman 1996], [Ackerman and Palen 1996], and [Ackerman et al. 1997]) and what the social requirements for such systems might be (e.g., the study of existing help systems in [Ackerman and Palen 1996]). However, we also believe to back up and consider how knowledge is stored and used in real work situations.

Ackerman and Halverson [Ackerman and Halverson 1998, Ackerman and Halverson 1999, Ackerman and Halverson 2000, Ackerman and Halverson 2002] examined the use of expertise in organizations, where expertise included both the use of documents and people in organizations. These papers presented findings from an analysis of a human resources hotline. Hotlines are particularly interesting places to study expertise seeking, because there is a constant, mostly repetitive flow of questions and information. The repetition makes the analysis easier, but there are enough varying questions to examine how non-routine queries are processed. The particular hotline for these studies answered personnel questions in a large computer company.

In these hotline studies, we determined that:

- The memories used by the participants were *simultaneously embedded* within several organizational, group, and individual processes. In the site, information was complexly distributed and occasionally overlaid with multiple uses. Moreover, the memories had *mixed provenance*: Sometimes the memory used was individual and private; sometimes it was group and public. For example, a call tracking record is a digital record of the call, which can later be accessed. The call tracking records compile a variety of statistics used at different organizational levels. In this way the call "memory" becomes a part of performance record for the person who handled the call, as well as information used at the group and organizational levels to plan. Thus, the same call record can have many different uses at different levels of organization.
- All of these memories must be used together seamlessly (or nearly so) to create an organizational product. The density and connectedness of memories used as resources can be remarkable. Within the organizational processes examined in the site, some information served as *boundary objects* [Star 1989]. As mentioned, while the representation is the same and the information looks the same, its meaning changes along with its users [Halverson 1995, Hutchins 1995]. For example, when verifying that a person is an employee of the company, a hotline agent knows only the "facts" given in the payroll database. The telephone agent knows none of the details of the payroll record's creation or maintenance; almost the entire context has been lost. She will not know whether there are problems with the record, such as the database not showing longtime temporary workers, or with employee's employment, such as probation. However, a hotline agent can use the payroll record in a satisfactory manner to determine the basic "fact" of employment. Removal of the detail and general agreement on a common-enough set of meanings enables the hotline agents to get their work done.
- To use information as a boundary object requires the loss of its contextualized information as it passes over the boundary. Those that need to use the information must expect this *decontextualization*. To reuse information later, a user must then *recontextualize* that information [Ackerman 1993, Braudel 1980, Oakeshott 1983, Schmidt and Bannon]. The information, if not supplied by the

same individual, must be re-understood for the user's current purposes. This can be a difficult matter, although people do it everyday in their work.

- Call records were not complete transcripts of everything said between hotline personnel and the caller. Instead, the agent who handled the call decided what would be the necessary information needed for subsequent reuse. To properly serve the re-user of information, the creator must properly project the consequences of the memory's later use; that is, they must determine the information's *trajectory*. Trajectory [Hutchins 1995, Strauss 1993] describes the path of an event; in this case, we mean it to be the likely trajectory as anticipated. The incentives for keeping information for later reuse appear to follow the assumed trajectory and its *projected consequences*. In the hotline, if an agent assumed that a call was routine and would not be referenced again, she had little incentive to write many details of the call. If that record must be reused in the future, the future user must deal with unanticipated consequences of the author's projecting the trajectory incorrectly.

Knowledge management largely restricts repositories to experience "objects" that are magically reusable, but it is more fruitful to consider expertise sharing as both *object* and *process*. What is of interest is both a memory artifact that holds its state and an artifact that is simultaneously embedded in many organizational and individual processes. The container metaphor implied by objectifying memory is easier to consider computationally, but it is extremely limited organizationally.

Recently, this line of research was extended to study an aircraft manufacturing hotline, available to help airline operators when they have significant repairs or problems with aircraft [Lutters 2001, Lutters and Ackerman 2002]. The study examined how hotline engineers balance timeliness with safety and reliability. In addition to understanding the context of the work and the organizational structures that ensure safety, Lutters was able to determine the role of the information, as both information object and as part of an information process, that is passed back and forth between the hotline and the airline engineers.

These field studies and technical studies have reinforced one another, enabling us to construct more organizationally viable systems. Throughout all of these studies, we have seen that by including the social-structural and social-relational aspects of an organization, we can foster more usable and useful knowledge management systems.

4.2 Finding expertise

The Answer Garden series of systems assumed that considerable knowledge existed in the heads of people. Indeed, the back-end of these systems required engines to find an expert or someone with suitable expertise. In this work, we have tried to augment the social-structural and social-relational aspects of an organization with systems, and our approach has been multi-pronged – combining repositories and networks. Again, our approach has been to determine how seeking expertise is done in natural settings through field-based studies, followed by with how best to support and augment expertise seeking through the construction of experimental systems. The following section surveys these two approaches.

4.2.1 Field studies of expertise finding

We have sought to understand expertise seeking through a set of social studies of expertise seeking in natural settings. In MacDonald and Ackerman [McDonald and Ackerman 1998], we examined how people sought others' expertise in a medium-sized software company. We called this software company MSC. MSC built a family of products to automate the back-end of doctors' and dentists' offices. Their systems were long-lived transaction systems, several of which had been in production for over two decades. They used a propriety software infrastructure and tools.

MacDonald and Ackerman found that expertise seeking in MSC could be analytically separated into three "phases": identification, selection, and escalation. These "phases" were often not separable or sequential in the everyday activity of the software engineers and others in MSC; however, they were separable enough to construct a system architecture based on them. Identification was the act of determining who might know the answer to a specific question. Selection was determining who was available or likely to provide the information. Escalation was the act of looking for additional people, perhaps crossing organizational boundaries or going to others that one might not normally consider. Each of these "phases" was necessary to obtain expertise. In the judgment of MacDonald and Ackerman, identification was the easiest to augment; selection and escalation could be augmented as well, but were more difficult. This analysis resulted in the system architecture of Expertise Recommender, described below.

4.2.2 Systems for finding expertise

MacDonald [McDonald 2000, McDonald and Ackerman 2000] constructed a system based on the field study reported in McDonald and Ackerman [McDonald and Ackerman 1998]. The system, called Expertise Recommender (ER), was designed to help people in MSC find others with the suitable program expertise to answer specific program questions. (See Figure 4.) ER's architecture assumed a number of general identification and selection heuristics (e.g., "find people nearby organizationally"), but also allowed site-specific and group-specific modules. In MSC, the programmers annotated their changes on line 10 of a modified program; thus, one ER module for MSC searched who had most recently changed a program. While the architecture was designed to be general across organizations, the field study findings suggested that the methods used for identification and selection were very local and contextualized. Therefore ER's specific finding heuristics also had to be very local and contextualized.

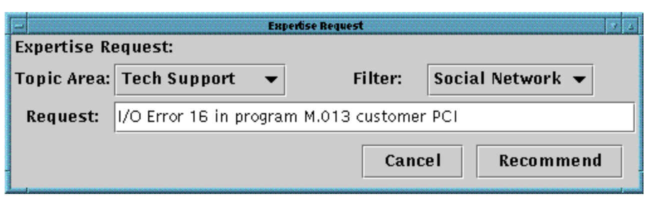


Figure 4: Expertise Recommender (ER)

4.2.3 Finding data

In addition to determining suitable systems architecture, we have also examined how to find the data to feed it. Our work has largely consisted of attempts to find what we call first-order approximations to measuring an organization's expertise network [Ackerman et al. 2002]. Because fully measuring the network is too time-consuming and costly (and often cannot be maintained adequately), we have looked for discount methods. While we hope to largely use the digital traces of one's work and identity, these traces must be bootstrapped with some measurement.

Our efforts, therefore, have been to find quick measures of what is important to know within a group and who knows what. We found that we could get participants to construct "Trivial Pursuit" questions that would measure critical success factors for a group or organization. Not only did participants enjoy being measured in such a way, they were also surprisingly willing to guess how others would do. Indeed we found that any 7 randomly selected software engineers or the 3 managers were able to estimate the group members' performances nearly as well as administering the test to everyone. More work will be necessary to determine whether this will be an approximation to the approximation and whether this straightforwardly extends to a very large company. Nonetheless, these first results were very encouraging.

4.3 Lightweight social spaces

One type of augmentation to an organization's sharing of expertise is to route queries to appropriate others with suitable expertise. Another type of augmentation is to create places or virtual spaces where people with suitable expertise congregate. Those with questions or those who wish to gain the expertise can then also go there.

There have been many successes within the use of computer-mediated communication systems where new forms of collaboration emerged [Sproull and Kiesler 1991]. A study of the Zephyr messaging system at MIT [Ackerman and Palen 1996] showed that chat or Instant Messenger-like systems could effectively be used for providing help. Furthermore, this study, as well as Ackerman et al. [1997], Muramatsu and Ackerman [1998], and Lutters and Ackerman [2002], showed many of the role, reward, and norm structures important to socially maintaining the place over time.

Zephyr is a heavily used Instant Messenger-like system created at MIT (though it was in use before IM). The Help Instance alone (one channel on Zephyr) receives over 30,000 messages per semester. The system is over eleven years old and has only discretionary use (i.e., no one is paid to answer questions on the system). The following is an example of Zephyr use:

```
Time: 06:27:32 Date: Thu Oct 14 93
From: College life is vastly overrated, according to US News and
World Report.  <elf>
Who wrote "Hallelujah!"?  Or is the author unknown?

Time: 06:28:27 Date: Thu Oct 14 93
From: Mobeus was two-faced <benjy>
If you're speaking of the Halleljuah chorus,
it is from Hayden's Messiah.

Time: 06:28:36 Date: Thu Oct 14 93
From: Kathy Talbott <shilla>
Handel, not Hayden
```

One should note that the answer was obtained in slightly more than a minute (at 6:30am). Furthermore, the public and visible nature of the questions and responses makes it possible to obtain correct and authoritative answers. A number of other social mechanisms, including reinforcement for displaying expertise in the MIT environment, contribute to obtaining correct answers as well.

Two systems that were especially constructed to facilitate knowledge transfer were Babble and Loops [Erickson et al. 1999]. Babble and Loops are two versions of a chat-like communication tool, originally designed to support small workgroups. Babble is a client-server based application in which typed messages are transmitted across a TCP/IP network, stored on a server and displayed to each client. Babble allows its users to engage in synchronous or asynchronous textual conversations, and provides visual feedback regarding who has recently participated in a conversation (see {Erickson, 1999 #16}). Loops moves the Babble experience onto a web-based platform, but maintains the features in Babble.

Babble and Loops look and feel like other forms of computer mediated communication (CMC); yet, neither is a bulletin board, a chat room system, a MUD, an email system, or a newsgroup. Babble and Loops merge the persistence and sequencing found in asynchronous bulletin boards with the immediacy and informality of MUDS and chat rooms. Loops adds more explicit shared bulletin board aspects – creating a space that is public, and semi persistent (i.e. changeable by any one). These combined features result in a blended synchrony user experience, in which interactions can shift naturally between synchronous and asynchronous modes depending on who is around and how actively they are participating.

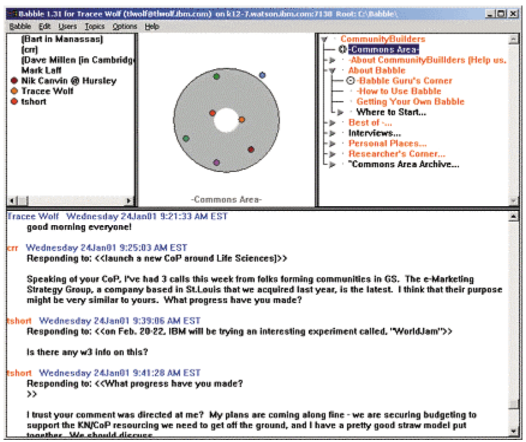


Figure 5: The Babble interface. From left to right, the top row of panes shows the user list, the social proxy, and the topic list. Below is the selected conversation.

This is possible because a major component of both Babble and Loops is the social proxy – a minimalist graphical representation of user activity. Seeing who is currently participating and the state they are in makes the social proxy a resource governing social behavior [Bradner, Kellogg, and Erickson 1999].

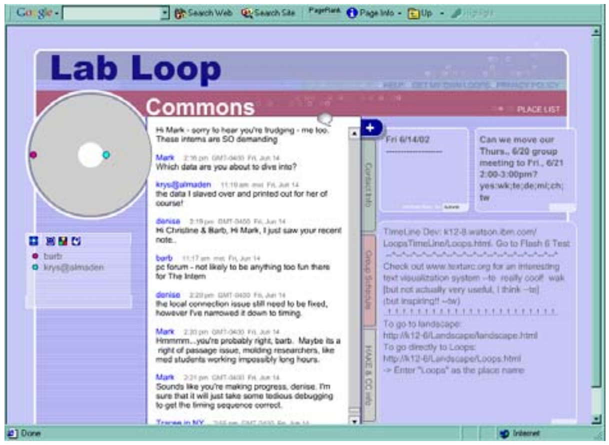


Figure 6: The Loops interface. Many of the elements are the same as Babble. The social proxy is on the far left with the user list below. The current conversational place is in the middle while the right side acts as bulletin boards.

Both Babble and Loops (Figures 5 and 6) display most of the same information, just in a different organization. This includes a list of all connected users; the social proxy; a list of conversations; and the content of the current conversation (i.e., the text). They use slightly different metaphors, so that in Babble, conversations are referred to as topics, while in Loops they are places where conversations occur. (In figure 6 the pull down in Loops that shows all the Places is hidden). Messages are appended to the bottom of the conversation pane and appear in the order posted.

Zephyr, Babble and Loops are examples of technically-created places in which people can gather to exchange information, ask particular types of questions, and share expertise with one another. Interestingly, Zephyr fosters relational ties as a second order effect of the expertise search. In Babble, which foregrounds these social interaction, we have seen new examples of expertise searching and sharing behaviors (e.g., waylay, as seen in [Bradner, Kellogg, and Erickson 1999]). We have also seen situations with information sharing as a side effect of the social relations in these social spaces. As such, these lightweight social spaces begin to fold back into expertise finding, augmenting the social network of an organization.

5 Conclusions and research challenges

In this chapter, we have demonstrated some new possibilities for knowledge management and sharing expertise that attempt to combine organizational and technical feasibility. These efforts assume the possibility of technical augmentation to the way an organization shares expertise either through a social network or a repository, but they attempt to do this augmentation in organizationally feasible ways.

This feasibility results from a consideration of the social-structural and social-relational aspects of an organization – critical dimensions of social capital. Not only are the organizational aspects described critical for organizational adoption and use, we hope that properly constructed systems can also further foster and promote a sense of social cohesion –social capital in its best light.

Above we showed our work in:

- *Augmenting repositories with social networks*. Incorporating an organization's social network into a repository makes the repository more organizationally robust and responsive. Some of the work incorporates mechanisms to ask people nearby (however measured), assuming that people who were more tied to the information seeker were more likely to respond.
- *Self-organizing expertise locators*. This work directly augments the social network of an organization to promote expertise sharing. The work also assumes that relational aspects (e.g., trust) are critical to effective use.
- *Communities that maintain and promote themselves as places*. This work attempts to create new social structures for an organization to foster expertise sharing.

In all of this work, then, we have assumed that incorporating the social-structural and social-relational aspects of social capital were critical to effective knowledge management and expertise seeking. All of the systems include or reconfigure an organization's social network, (social-structural), provide incentives and inculcate trust (social-relational), and lead to shared understanding and mental models (social-cognitive). While first-generation knowledge management approaches, based on individual, cognitively-based technologies (e.g., incorporating an Intranet with information retrieval search), we believe that significant benefits will accrue only with understanding the need for social capital and incorporating its dimensions into all types of knowledge management technologies.

We are currently extending this work to consider how to form new connections among parts of a social network. Even new possibilities exist. One can imagine creating distributed coalitions, where social subnetworks of two organizations, institutions, or voluntary associations were joined, perhaps internationally. As well, one can imagine augmenting all of this social activity with agent clusters to find, broker, and reward those in the subnetworks.

In our current work, which extends Ackerman and Halverson [Ackerman and Halverson 2002], we are considering how to:

- Design within a social-technical *co-design space.*
- How to create new assemblages as *resources* for users [Halverson and Ackerman 2003, Halverson 1995, Hutchins 1995].

We expect this work to give us significant insights into designing new and organizationally feasible systems.

Acknowledgements

Babble and Loops are projects of IBM. Other projects described in this chapter have been funded, in part, by grants from National Science Foundation (IRI-9702904 and IRI-0124878), the UC Irvine/NSF Industry/University Cooperative Research Center at the Center for Research on Information Technology and Organizations (CRITO), NASA, the University of California MICRO program, Interval Research, Quality Systems, and the MIT/Project Oxygen partners.

This work has benefited from far too many conversations over the last decade to even mention. We especially want to thank our collaborators, Wendy Kellogg and Tom Erickson. David McDonald, Wayne Lutters, and Jack Muramatsu were members of the team at UC Irvine. We would also like to thank Volker Wulf and Marlene Huysman for their continued interest and support in this work.

References

Ackerman, Mark S. 1993. Answer Garden: A Tool for Growing Organizational Memory. Massachusetts Institute of Technology, Ph.D. Thesis.

Ackerman, Mark S. 1994. Augmenting the Organizational Memory: A Field Study of Answer Garden. *Proceedings of the Conference on Computer-Supported Cooperative Work (CSCW'94)* : 243-252.

Ackerman, Mark S. 1996. Definitional and Contextual Issues in Organizational and Group Memories. *Information Technology and People*, 9 (1) : 10-24.

Ackerman, Mark S. 1998. Augmenting Organizational Memory: A Field Study of Answer Garden. *ACM Transactions on Information Systems*, 16 (3) : 203-224.

Ackerman, Mark S. 2000. The Intellectual Challenge of CSCW: The Gap Between Social Requirements and Technical Feasibility. *Human-Computer Interaction*, 15 (2-3) : 179-204.

Ackerman, Mark S. 2001. The Intellectual Challenge of CSCW: The Gap Between Social Requirements and Technical Feasibility. In *HCI in the New Millennium*. Edited by J. Carroll. New York: Addison-Wesley.

Ackerman, Mark S., James S. Boster, Wayne G. Lutters, and David W. McDonald. 2002. Who's There? The Knowledge Mapping Approximation Project. In *Sharing Expertise: Beyond Knowledge Management*. Edited by M. S. Ackerman, V. Pipek and V. Wulf. in press. Cambridge, MA: MIT Press.

Ackerman, Mark S., and Christine Halverson. 1998. Considering an Organization's Memory. *Proceedings of the Computer Supported Cooperative Work (CSCW'98)* : 39-48.

Ackerman, Mark S., and Christine Halverson. 1999. Organizational Memory: Processes, Boundary Objects, and Trajectories. *Proceedings of the Proceedings of the IEEE Hawaii International Conference of System Sciences (HICSS'99)*

Ackerman, Mark S., and Christine Halverson. 2000. Re-Examining Organizational Memory. *Communications of the ACM*, 43 (1) : 58-63.

Ackerman, Mark S., and Christine Halverson. 2002. Organizational Memory: Processes, Boundary Objects, and Trajectories. Working paper.

Ackerman, Mark S., Debby Hindus, Scott D. Mainwaring, and Brian Starr. 1997. Hanging on the 'wire: A field study of an audio-only media space. *ACM Transactions on Computer-Human Interaction*, 4 (1) : 39-66.

Ackerman, Mark S., and Thomas W. Malone. 1990. Answer Garden: A Tool for Growing Organizational Memory. *Proceedings of the ACM Conference on Office Information Systems* : 31-39.

Ackerman, Mark S., and David W. McDonald. 1996. Answer Garden 2: Merging Organizational Memory with Collective Help. *Proceedings of the ACM Conference on Computer-Supported Cooperative Work (CSCW'96)* : 97-105.

Ackerman, Mark S., and Leysia Palen. 1996. The Zephyr Help Instance: Promoting Ongoing Activity in a CSCW System. *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'96)* : 268-275.

Ackerman, Mark S., Vokmar Pipek, and Volker Wulf. 2002. *Sharing Expertise: Beyond Knowledge Management*. Cambridge, MA: MIT Press.

Allen, Thomas. 1977. *Managing the Flow of Technology*. Cambridge, MA: MIT Press.

Bradner, E., W. Kellogg, and T.. Erickson. 1999. The Adoption and Use of "Babble": A Field Study of Chat in the Workplace. *Proceedings of the 6th European Conference on Computer Supported Cooperative Work (ECSCW'99)* : 139 - 158.

Braudel, Fernand. 1980. *On History*. Chicago: University of Chicago Press.

Cohen, Don, and Laurence Prusak. 2001. *In Good Company: How Social Capital Makes Organizations Work*. Boston: Harvard Business School Press.

Erickson, Thomas, David N. Smith, Wendy A. Kellogg, Mark Laff, John T. Richards, and Erin Bradner. 1999. Socially translucent systems: social proxies, persistent conversation, and the design of "babble". *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'99)* : 72-79.

Feldman, Daniel C. 1984. The Development and Enforcement of Group Norms. 9 (1) : 47-53.

Garfinkel, Harold. 1967. *Studies in Ethnomethodology*. Englewood Cliffs, NJ: Prentice-Hall.

Goffman, Erving. 1961. *The Presentation of Self in Everyday Life*. New York: Anchor-Doubleday.

Grudin, Jonathan. 1989. Why groupware applications fail: problems in design and evaluation. *Office: Technology and People*, 4 (3) : 245-264.

Halverson, Christine, and Mark S. Ackerman. 2003. Yeah, the Rush ain't here yet -- Take a break. *Proceedings of the IEEE Hawaii International Conference of System Sciences (HICSS'03)*

Halverson, Christine A. 1995. Inside the Cognitive Workplace: New Technology and Air Traffic Control. University of California, San Diego, Cognitive Science Department, Ph.D. Thesis.

Hutchins, Edwin. 1995. *Cognition in the Wild*. Cambridge, MA: MIT Press.

Huysman, Marleeen. 2003. Design requirements for knowledge sharing tools; A need for social capital analysis. In *Social Capital and Information Technology*. this volume. Cambridge: MIT Press.

Lesser, Eric, and Laurence Prusak. 2000. Communities of Practice, Social Capital and Organizational Knowledge. In *Knowledge and Communities*. Edited by E. L. Lesser, M. A. Fontaine and J. A. Slusher. 123-132. Boston: Butterworth-Heinemann.

Lesser, Eric L., Michael A. Fontaine, and Jason A. Slusher. 2000. *Knowledge and Communities*. Boston: Butterworth-Heinemann.

Lutters, Wayne G. 2001. Supporting Reuse: IT and the Role of Archival Boundary Objects in Collaborative Problem Solving. PhD Dissertation,

Lutters, Wayne G., and Mark S. Ackerman. 2002. Achieving Safety: A Field Study of Boundary Objects in Aircraft Technical Support. *Proceedings of the ACM Conference on Computer-Supported Cooperative Work (CSCW'02)* : accepted for publication.

McDonald, David W. 2000. Supporting Nuance in Groupware Design: Moving from Naturalistic Expertise Location to Expertise Recommendation. PhD Dissertation, University of California, Irvine.

McDonald, David W., and Mark S. Ackerman. 1998. Just Talk to Me: A Field Study of Expertise Location. *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW '98)* : 315-324.

McDonald, David W., and Mark S. Ackerman. 2000. Expertise Recommender: A Flexible Recommendation System Architecture. *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW'2000)* : 231-240.

Muramatsu, Jack, and Mark S. Ackerman. 1998. Computing, Social Activity, and Entertainment: A Field Study of a Game MUD. *Computer Supported Cooperative Work: The Journal of Collaborative Computing*, 7 (1) : 87-122.

Nahapiet, Janine, and Sumantra Ghoshal. 1998. Social Capital, Intellectual Capital, and the Organizational Advantage. *Academy of Management Review*, 23 (2) : 242-266.

Oakeshott, Michael. 1983. *On History and Other Essays*. Totowa, NJ: Barnes and Noble Books.

Orlikowski, Wanda J. 1992. Learning from Notes: Organizational Issues in Groupware Implementation. *Proceedings of the Computer Supported Cooperative Work (CSCW'92)* : 362-369.

Schmidt, Kjeld, and Liam Bannon. Taking CSCW seriously: supporting articulation work. *Computer-Supported-Cooperative-Work-(CSCW)*, 1 (1-2) : 7-40.

Sproull, Lee, and Sara Kiesler. 1991. *Connections: New Ways of Working in the Networked Organization*. Cambridge, MA: MIT Press.

Star, Susan Leigh. 1989. The Structure of Ill-Structured Solutions: Boundary Objects and Heterogeneous Distributed Problem Solving. In *Distributed Artificial Intelligence*. Edited by L. Gasser and M. Huhns. 37-54. San Mateo: Morgan Kaufmann.

Stewart, Thomas A. 2001. *The Wealth of Knowledge: Intellectual Capital and the Twenty-first Century Organization*. New York: Doubleday.

Strauss, Anselm. 1991. *Creating Sociological Awareness: Collective Images and Symbolic Representations*. New Brunswick: Transaction.

Strauss, Anselm L. 1993. *Continual Permutations of Action*. New York: Aldine de Gruyter.

Suchman, Lucy A. 1987. *Plans and Situated Actions: The Problem of Human-Computer Communication*. New York: Cambridge University Press.

Teasley, S., L. Covi, M. Krishnan, and J. Olson. 2000. How does radical collocation help a team succeed? *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW2000)* : 339-346.

Wenger, Etienne. 1998. *Communities of practice : learning, meaning, and identity*. New York: Cambridge University Press.