Facilitating Collaborative Learning in Distributed Organizations

Jim Greer, Gordon McCalla, Vive Kumar, Jason Collins, Paul Meagher

ARIES Laboratory, Department of Computer Science, University of Saskatchewan, Canada

Abstract

Increasingly, organizations are geographically distributed with activities coordinated and integrated through the use of information technology. Such organizations face constant change and the corresponding need for continual learning and renewal of their workers. In this paper we describe a system called PHelpS (Peer Help System) that facilitates workers in carrying out such "life long learning". When a worker runs into difficulty in carrying out a task, PHelpS provides a list of other workers who are ready, willing and able to help him or her. The worker then selects a particular helper with PHelpS supporting the subsequent help interaction. The PHelpS system acts as a facilitator to stimulate learning and collaboration, rather than as a directive agent imposing its perspectives on the workers.

Keywords—peer help, collaborative learning, user modelling, workplace training

Introduction

A successful modern organization is characterized by workers who feel empowered and take ownership and pride in the things they do. It has a climate where employees at all levels help one another to meet the organization's mission. It fosters and rewards continual learning, encourages cross-training and collegial ties, and distributes the organizational knowledge across the entire workforce. But a successful modern organization doesn't just happen spontaneously. It develops as a result of organizational change where a management-created climate encourages the growth of grass-roots communities of learners and actively supports collaborative learning efforts.

One common component of collaborative learning is the "informal peer-help network". This is a social network of a worker's colleagues and acquaintances who are consulted for help and advice about work matters. Informal peer-help networks exist in every organization. They constitute a crucial component of training and acclimatization for new employees and represent an essential element of shared organizational memory. The goal of our project is to use technology to support collaborative learning and the expansion of informal peer-help networks in large distributed organizations, in particular those where information technology plays a central role.

In order to effectively use modern information technology a worker must possess both lower-level procedural and higher-level problem-solving and judgement skills. On the job performance support (through a good on-line help system or task checklists) can assist workers to overcome many simple impasses. Nevertheless, no matter the degree of training or skill, workers using information technology often need to request just-in-time help from someone in their informal peer-help network. More experienced users will make less frequent but more complex help requests, while novice users tend to make more frequent and normally less complex help requests.

The specific objective of our project is to provide assistance to workers who are learning to make effective use of a management information system in their workplace. This involves activities that range from training to performance support to just-in-time help. The approach we have taken is to develop a task-oriented performance support system that is tightly coupled with a peer help system (dubbed PHelpS). There are two novel aspects of the PHelpS approach. The first involves the use of a task-oriented performance support system to index help requests and user knowledge. The second involves knowledge-based support for locating a peer somewhere in the distributed organization who is ready, willing, and able to provide help when needed. Combining these aspects produces an effective environment for delivering just-in-time help. It also creates a fertile environment in which to study collaborative learning and organizational change. Finally, the necessity to maintain knowledge profiles of potential helpers has produced many interesting research challenges in the domain of user and learner modelling.

This paper describes the PHelpS approach, concentrating on its strengths in supporting collaborative learning through peer help and speculating on its potential to effect organizational change.

Background

Constant et al (Constant et al., 1996) characterize and describe "help exchanges" within organizations. They indicate that people prefer to exchange help by way of collegial ties, which develop with physical proximity, group membership, prior relationships and demographic similarity. In contrast to the strong ties of personal friendships within an organization, they advocate the development of weak ties among wider collections of workers so that informal peer-help networks can be expanded. This notion is compatible with Wenger's communities oflearners (Wenger, 1996), where people who share learning goals within an authentic learning environment can develop ties that reinforce learning outcomes.

Developing collegial ties within a distributed organization is a challenge, since co-workers may be strangers separated by space and time. These separations pose similar challenges to those impeding the development of collegial ties in distance learning environments. Computer support for nurturing such collegial ties can be helpful. Furthermore, the provision of peer-help and peer tutoring has the positive side effect of strengthening the knowledge and confidence of both the person being helped and the helper or tutor (Palthepu et al., 1991; Slavin, 1990).

On the issue of providing help for workers, Carroll (Carroll, 1990) demonstrates the ineffectiveness of de-contextualized help and the effectiveness of task-oriented help in learning about computer and information systems. His claims are sympathetic with the emphasis placed on social context in the situated learning community. Interestingly, in workplace settings where tasks and duties are fairly well specified, it becomes relatively straightforward to define and represent task models that can be used to encourage situated learning. There is a definite connection between task models and the potential for developing collegial ties. Weak ties develop in the context of carrying out workplace tasks (Constant, 1996). Thus, understanding the nature of tasks has a dual benefit of better understanding and addressing help needs and better supporting the development of collegial ties. This support can lead to the self reinforcing benefit of increased peer collaboration, more collaborative learning in the organization, and may eventually result in positive changes in the way workers cooperate with one another and how they think about and carry out their tasks (Hammer, 1996).

Client Group and Prototype Deployment Site

The PHelpS system (Collins et al., 1997; McCalla et al., 1997) has been developed, tested and deployed in the context of the Correctional Services of Canada (CSC) as part of a staff training initiative sponsored by the Canadian TeleLearning Network of Centres of Excellence program. CSC is a national distributed organization with some 11000 workers in some 281 different locations. PHelpS prototypes have been tuned to suit the needs of workers in the Prairie Regional Psychiatric Centre (RPC), one of the facilities of CSC. A large CSC-wide information system called the Offender Management System (OMS) has been recently introduced within CSC and almost all workers are expected to make significant use of this system in their daily activities. PhelpS functions "on top of" OMS to provide performance support and to facilitate peer help with tasks related to OMS. While the PhelpS approach is not limited to OMS, CSC or RPC, the examples of its use presented in this paper are drawn from our experiences with its deployment at RPC.

The PHelpS Approach

The PHelpS system is being developed to deliver a type of just-in-time training by providing help, especially peer help, in the context of completing real tasks. PhelpS assumes that tasks are well defined and represented as task hierarchies within the system. Various system modules interact with the task hierarchies. The architecture of the PhelpS system is shown in Figure 1. The architecture consists of two distinct modules: the Personal Assistant, which provides performance support, case-based help and peer help during task execution; and the Knowledge Update module, which permits prospective peer helpers and users to inspect and update the system's models of themselves. These components utilize three knowledge bases: the Task Hierarchy repository, the Learner/Helper Model repository, and the Help Case repository. Each of these is briefly explained in the following sections.

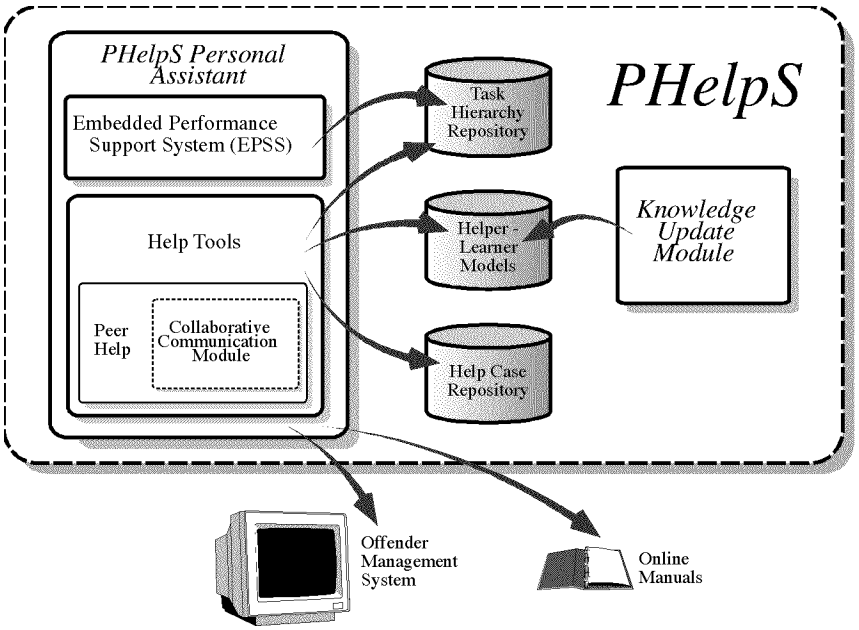

Figure 1: The PhelpS System Architecture

Task Hierarchies

It is possible in many organizations to capture the essentials of common tasks carried out by workers in a hierarchical structure we call a task hierarchy. Knowledge engineering in PHelpS requires a phase of task hierarchy construction, where typical tasks are analyzed, processes for completing the tasks are agreed upon, and the task process steps are codified in a hierarchical structure. These task hierarchies become the fundamental data structure underlying the PHelpS approach.

A fragment of a typical task in CSC (the Escorted Temporary Absence (ETA) task) is illustrated in the upper panel of Figure 2. The ETA task consists of a set of (approximately 80) steps corresponding to specific subtasks that must be completed in order that an offender may be granted an escorted temporary absence from the correctional facility (for example, to attend a funeral of a family member). Many of these subtasks involve referring to or updating the OMS database. While policies and procedures for tasks like the ETA task exist within CSC, there were no existing detailed official task hierarchies for any tasks when our research began. Our first steps involved doing task analyses of various tasks (in the sense of Shepherd 1995) and constructing a small repository of representative task hierarchies.

Personal Assistant

The Personal Assistant is an embedded performance support system (EPSS) (McGraw, 1994) with a suite of embedded help tools, including a peer help system. The Personal Assistant provides checklists of actions that correspond to specific routine tasks commonly performed within an organization. The Personal Assistant provides the means through which users can inspect and perform various task-oriented operations. A task hierarchy viewer helps the worker to use task hierarchies as checklists in recording steps completed within their tasks (see the checkmarks in the upper panel of Figure 2). In the CSC context, this viewer is attached to the OMS interface (see the lower panel of Figure 2) and serves a dual role as a reminder of the current and next steps in the task and as a hyper-linked shortcut to relevant OMS screens.

When a step in the task checklist is causing difficulty, the user can mark the step with a "question mark" and then can request various modes of help. Help options range from pop-up explanations of checklist descriptors, to context-specific help from online manuals, to case-help browsing where relevant prior help cases can be viewed, to help with locating a human peer who can provide knowledgeable advice. Peer help is the focus of this paper.

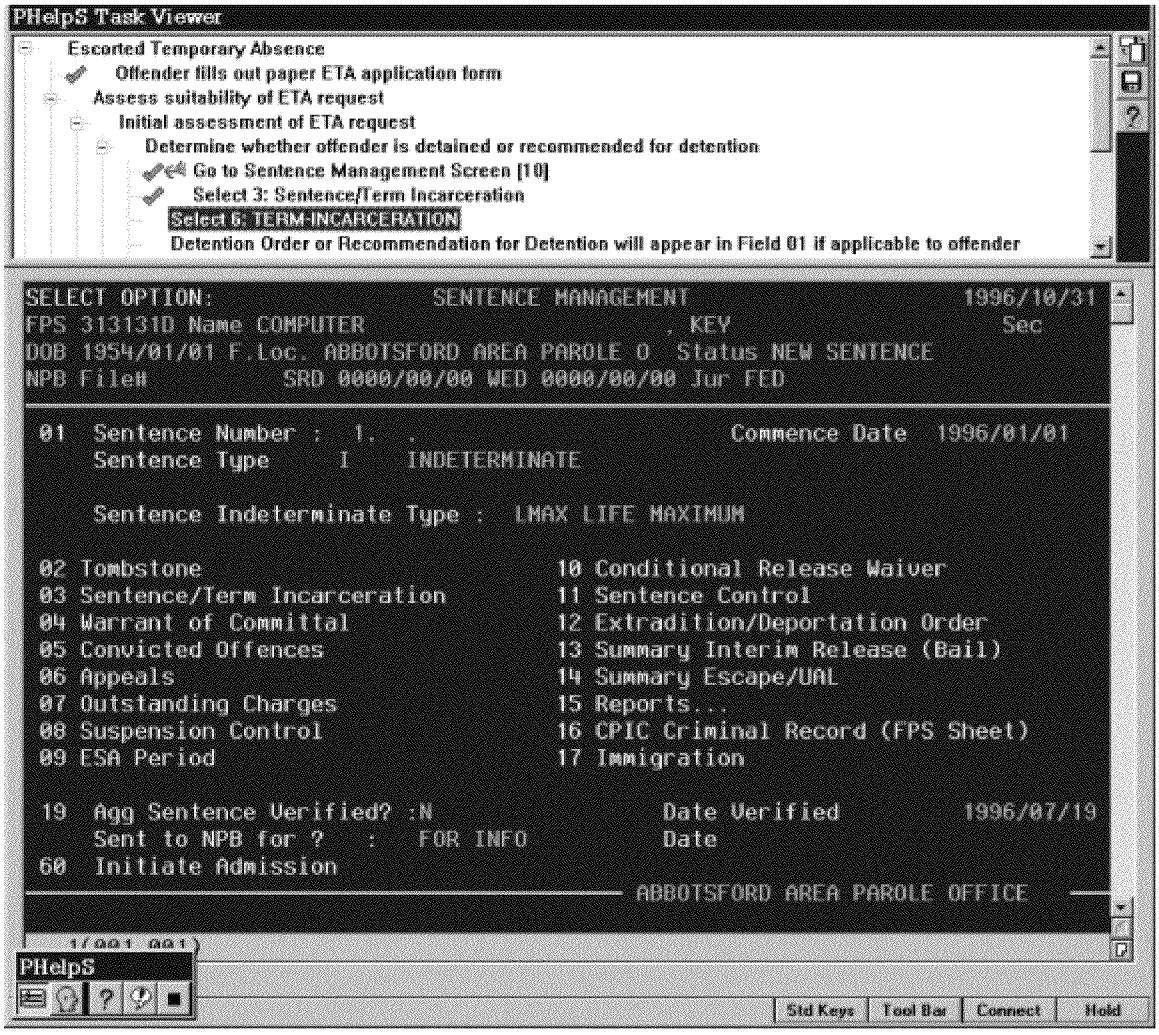

Figure 2: The PHelpS Personal Assistant Surrounding OMS

Peer Selection Algorithm

Selecting an appropriate peer helper is a knowledge-intensive process. The PHelpS system assists with this process by matching the task at hand with "knowledge profiles" of candidate peer helpers, bearing in mind constraints on availability and willingness to provide help. Ideally, every worker in the organization is considered to be a candidate helper, and as such every worker should have a knowledge profile maintained by the system. In practice a set of volunteer peer helpers is a more likely prospect, which is the current situation at RPC. The candidate helper's knowledge profile is essentially an overlay on the various task hierarchies recording the degree of capability the person has with each important task and subtask.

Within the ETA task hierarchy, 15 of the steps are considered important subtasks from a helper selection perspective. The other finer-grained steps (such as knowing how to press the "esc" key or to enter an offender's name) are considered irrelevant when computing a helper's knowledge profile. Thus, skill with these 15 important subtasks would be recorded in each helper's knowledge profile for the ETA task. Factors influencing the system's belief about the helper's skill include the number of times the specific subtask has been completed in the recent past, the number of times he or she has given help on the subtask in the past, and the number of times this help was useful or not useful to the worker requesting help. These counts are combined into a composite numeric measure of helper capability for each subtask.

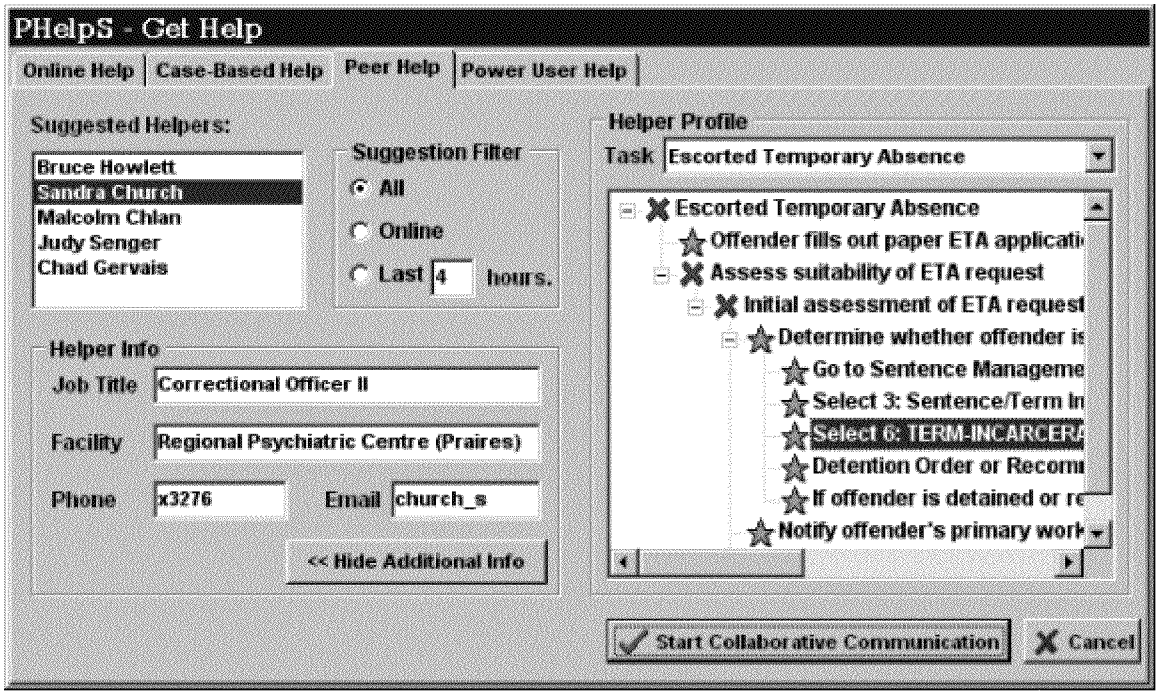

Figure 3: Requesting peer help in PHelpS

When a worker requests help on a step in the checklist, the helpers who have sufficient capability on the nearest important subtask (and neighbouring subtasks in the task hierarchy) are selected from the database and considered as part of the pool of candidate helpers. This pool of candidate helpers is narrowed to a short list by considering such factors as availability (whether or not the candidate helper is at work and on line) and willingness (whether or not the candidate helper is willing to be disturbed or has handled enough help requests for one day). Figure 3 presents one such scenario where the worker has requested peer help with respect to the task step highlighted in the right panel of Figure 3. The system has responded with a short list of five potential helpers. It is left to the person requesting help to make a final selection of a peer helper from this short list. Final worker control is an important attribute in making the peer help system both robust and acceptable to the workers.

In order to facilitate the choice of a helper, it is possible for the worker to inspect certain aspects of each potential helper's user Model (Collins, 1997). For example, the helper's job title, work location, linguistic preferences, and contact information are always available. At the discretion of the helper, other information may be made available to those seeking help. Data such as gender, age, seniority, helping preferences, other credentials, and the details of the helper's knowledge profile may also be viewed by the worker needing help (in the "Helper Info" panel of Figure 3), but only if the potential helper so desires. Keeping such user models of all workers raises issues of privacy and management control, as discussed in the conclusion.

User Model Acquisition and Maintenance

Each helper's user model contains information ranging from the task-specific (eg. the knowledge profiles showing the tasks the peer helper can perform and the level of capability in carrying out important steps in these tasks), to intermediate knowledge-specific features (eg. credentials, courses completed, documents read), to the very general (eg. the peer helper's age, gender, job title, linguistic preference, current login status, the number of times the peer helper has provided help, etc.). Information in a helper model changes over time, ranging from fine-grained temporal intervals (eg. minutes or hours for login status, or to track the particular task step being carried out), through longer term periods (eg. days or weeks for learning a new task, months or years for changing position in the organization, etc.).

Initial knowledge acquisition of the helper models is done through the completion of a questionnaire by workers who wish to be on the peer helper roster. Once initially stocked, however, the helper models must be maintained, i.e. kept up to date, as changes occur over time. This is the task of the Knowledge Update module.

For time varying information, such as login status, work schedules, or number of times the peer helper has dispensed help, user model maintenance is straightforward, since this information can be gleaned by the system. It is more difficult, however, for changes in the task-specific information in the knowledge profiles to be tracked. Nevertheless, the Knowledge Update module does provide automated support for such maintenance. Since each user is also a potential helper, as users carry out various tasks and use the task checklists, the system increases its confidence in their knowledge of the task. Thus, knowledge profiles can be updated by using information from the EPSS checklists. In addition, each time a helper is called upon for assistance, the number of times helped is automatically updated. If the user requesting help finds the interaction helpful, the helper's ability score is increased.

Despite these automatic aids, the ultimate responsibility for managing knowledge profiles is left in the hands of the users. We are not the first to suggest that users should help maintain their own user models (Paiva, 1994), but in order for this to work, the context must be right for them to do so. In PHelpS, users are frequently in situations where they can inspect their own and their colleagues' user models. This is supported by the continual updating of checklists by workers as they carry out tasks, by the inspection of the models of other workers during helper selection, and by the access to their own and their helper's user models during help sessions. Thus, users are constantly confronted with user model inaccuracies. They should also want to take action to correct these inaccuracies, so that (i) when they need help, it can be targeted appropriately to take into account their own strengths and weaknesses; and (ii) when others contact them for help, they will only be contacted on topics about which they know something. In short, an inaccurate user model will cost a user time both when seeking help and when being sought for help.

One final use of the Knowledge Update module is in annotating task hierarchies. Decisions made while knowledge engineering task hierarchies arise from consensus among experienced end-users, and thus evolve over a period of time. As task hierarchies are evolving to their final form, users are invited to annotate various subtasks in any task hierarchy and other users are able to browse these annotations. Subtasks within task hierarchies are subject to ongoing change as regulations or policies change. Annotations in task hierarchies are one means of identifying the need for change, and also in helping to manage change.

Experiments are being conducted to discover if, in fact, these assumptions about users and user models are reasonably accurate in the work environment at the Regional Psychiatric Centre of CSC. These include studies of PHelpS in classroom use during training, and deployed on workers' own workstations for use in getting on the job help.

Communication Support

As soon as a user selects a peer helper, a dialogue is initiated by the person needing help. This dialogue can happen off-line (eg. by telephone or personal visit) or on-line, depending on urgency and the predilections of the two people involved. In the current PHelpS version, an on-line dialogue between worker and peer helper is a mediated chat session with some facilities for knowledge sharing which we call PHelpS Talk. PHelpS provides context-based message templates to the user who can fill-up the template and send it off to the peer helper for either synchronous or asynchronous communication. Both helper and user can share the current task hierarchy and use it as an object of discussion during communication, as well as access each other's knowledge profiles and checklists (if permission of each user is given).

The dialogue exchanged through the PHelpS Talk Collaborative Communication facility is trapped for each case and becomes fodder for a case-based help facility. Capturing and indexing help cases is supported by the task-hierarchy knowledge representation underlying PHelpS. Each completed help case is examined by a moderator, edited to remove personal information, indexed to the appropriate subtasks where it might be applicable, sometimes generalized to include other information not in the original help dialogue, and finally committed to the Help Case repository.

This “frequently undertaken interaction” (FUI) case base is an extension of the “frequently asked question” (FAQ) idea in that the entire interaction

(rather than just a single question and answer) can be accessed and then browsed by workers for hints as to their current problem. Unfortunately, in situations where the Collaborative Communication module is bypassed in favour of telephone communications or personal visits, it becomes impossible to automatically capture help cases, although a manual transcription could be adapted by the moderator.

We have been re-implementing the Collaborative Communication facility in PHelpS to incorporate Microsoft NetMeeting software. In this environment, communication can include audio and video conferencing as well as application sharing and chat facilities. Within CSC, there are network challenges that might preclude audio and video conferencing at this time and there are serious security and accountability challenges that might preclude application sharing as well. Nevertheless, other organizations might be better suited to this kind of complete collaboration support environment for peer help. The PHelpS approach in conjunction with a point-to-point collaboration tool makes a strong partnership.

Presently PHelpS supports only passive collaboration where the system does not take an active part in supporting/enhancing collaboration. Such efforts fall under the realm of CSCW techniques. We are in the process of examining how the system might be expanded to support active collaboration where the system tries to understand the needs of individual workers based on their models, to provide appropriate feedback and structured communication media and, finally, to drive peer collaboration towards a preferred model of collaboration.

Informal usability testing

One recent usability study was conducted on PHelpS at RPC in one of the OMS training courses. The goal of this experiment was to provide hands-on training with OMS and PHelpS together and to test the effectiveness of PHelpS. The study consisted of two sessions with a total of four trainees, a small but representative group. The first session lasted for 75 minutes introducing OMS and PHelpS to the trainees. Realistic simulated tasks were given to the trainees to perform during the next 75 minutes. The trainees were in the training room and the helpers were on duty performing their day-to-day activities. The experimenters recorded the proceedings using audio (telephone conversations and debriefing), video (of the entire proceedings), trickle files (of the keystrokes) and "observation notes".

Lessons Learned

Analysing the collected data resulted in a number of useful observations that could help direct future developments in PHelpS.

- All the trainee workers used the PHelpS task hierarchy checklists. Three different types of checkmarking modes were observed: serial, burst and random. In serial mode, the workers checkmarked a task step immediately after performing it. In burst mode, the workers checkmarked a series of task steps after performing many task steps. In random mode, the workers checkmarked task steps only when they required peer assistance. Random mode checkmarking seems to be very common. This points out that the workers perceive PHelpS as a help tool that is best used only when they are seeking peer help.

- During the experiment, there were seven help requests from the trainee workers leading to seven help sessions, out of which five were successful. The failure of the remaining two sessions were attributed to the failure to communicate the full context of the help requests. In one help session it took several minutes to establish the correct context and only a few minutes to solve the problem. Some excerpts from this help session are given below.

Trainee: Yeah, but do I need, like, it says, schedule six full-time participants. Do I need the names of sixfull time participants?
Helper: Do you have the names of the people?
Trainee: Ahh, not so far. No.
Helper: Oh, OK.
Trainee: So, OK there is a scheduled program in the system already right. What, what is the program called?

…(the trainee leads the helper astray and the context is not mutually understood>
…(5 minutes elapsed)
Helper: They never made any mention of a list of guys or anything like that?
…(the helper tries to bring the trainee back to the correct context)
…(3 minutes elapsed)
Helper: So they want you to assign a guy, assign 6 people to the relationships program.

Trainee: Ah. Schedule. Schedule 6 full time participants to attend the relationships program.

Helper: Oh. OK. ahhh (cough). Go back to the main menu.

…(problem is identified and the helper proceeds with the correct help.)

- All trainee workers preferred to choose "a knowledgeable peer" and were satisfied to choose the person suggested by the system, even though they were sometimes strangers. In addition, the workers prefer to ignore the job hierarchy as long as the helpers were willing to collaborate. In one of the help sessions, a worker picked a third ranked helper from a list of helpers suggested by the system. Eventually that help session turned out to be a failure because the helper had little expertise in that particular context.

Discussion

We observed some very encouraging, but also thought provoking feedback about the PhelpS approach during our experimentation at the Regional Psychiatric Centre. Learners in post hoc debriefing sessions made comments such as

"PHelpS is very user-friendly ... like step-by-step processing, fully knowing what to do next … hierarchy is good"

"... getting the context right is tough"

"hands-on approach is good ... some staff have more knowledge on certain topics ... it is better to contact them when stuck"

"... being able to help someone else is helping you"

"I will contact a knowledgeable stranger to a less knowledgeable friend"

"Looking forward to seeing it (PHelpS) on the units"

We thus believe that the PHelpS approach, if widely deployed, would have significant promise. We are also aware that there are some concerns that have to be addressed, including

- how sophisticated the task hierarchies need to be to allow the selection of an appropriate peer helper and to capture a wide variety of tasks,
- how sensitive the matching algorithm needs to be to identify appropriate peer helpers,
- how change in tasks and workers capability in carrying out tasks can be handled,
- how communication between helper and the learner can be facilitated,
- how additional kinds of help can be incorporated,
- whether wide spread acceptance of PHelpS can be gained by both workers and management,
- whether PHelpS, once deeply ingrained in an organization, leads to abuse of workers' privacy, and is misused to monitor workers' efficiency and effectiveness.

Task hierarchies are central to all activities involved in PHelpS. There are approximately twenty high-level tasks that are carried out at RPC. The current PHelpS prototype captures only two of these high-level tasks, with each task comprising about two hundred task steps. It took about one person month to engineer each of these two tasks. It is our belief that the rest of the tasks can also be engineered in a similar time frame.

There are three major maintenance concerns that need to be addressed with respect to the task hierarchies. To start with, the engineered task hierarchies need to be verified for their accuracy. Second, the task hierarchies need to be updated from time to time to ensure consistency of the hierarchies across the different units within a CSC organisation, across the different CSC organisations in Canada and across the different user models that are maintained in PHelpS. Third, we are trying to identify ways to represent tasks that cannot be naturally represented in hierarchies. This would help us generalise the design of the peer help approach and replicate PHelpS-based peer support for different organisations with different work cultures from that at RPC.

We would also like to explore enhancements to the PHelpS system to incorporate more facilities to handle change. We think that belief revision techniques from artificial intelligence could be adapted to significantly improve the ability to keep knowledge profiles up-to-date in the face of changes in tasks and workers' understanding of these tasks. We are also actively experimenting with augmenting the other kinds of help the system can provide, including more sophistication in the FUI Help Case subsystem, and more refinement in online "canned" help to capture whether such help is at the technical or policy levels, thus providing the worker with further choice as to the kind of help he or she

feels is needed in the current situation. All such help is made more coherent and more useful by being indexed through the current task hierarchies to ensure its relevance to current worker goals (Carroll, 1990). A final area we are currently exploring for system enhancement is improving communication between the helper and the worker being helped. As mentioned above, we are incorporating advanced communication software such as NetMeeting. We also are working on expanding the role of the system during the help session by allowing it to actively intervene in and influence the interaction (by making use of the information in the user models of each participant in the session). The system itself becomes a learning collaborator in the interaction in the manner of Chan and Chou (Chan et al., 1995). A big remaining communications problem is coming up with techniques that help establish mutual context at the beginning of a help session (beyond sharing user profile information).

Another major concern for the success of PHelpS is the motivation of the workers to support each other. Peer help, formal or informal, is a common trend and is being encouraged in corporate cultures. PhelpS interactions are considered informal in the sense that the workers choose their helpers without being absolutely explicit as to what kind of help they need and without knowing much about the candidate helpers. PHelpS attempts to provide structured peer support by collecting information about the workers and helpers with respect to the task hierarchies, their work habits and other related information. The workers know that there are implicit rewards associated with peer help. In addition, the organisation should come up with explicit rewards to encourage peer support. For instance, RPC has recognised a group of "power users" who volunteered to work under a new initiative called the "peer-mentor program" within RPC. Such corporate recognition can be motivating for potential peer helpers.

With respect to privacy issues, the individual user models employed by PHelpS to help workers, could also be used by management to monitor workers, perhaps to the workers' perceived or actual disadvantage. Hopefully, management would be deterred from such monitoring by the likely result that workers would immediately cease using PHelpS at the first sign that their privacy was being violated, and thus all the potential advantages of the PHelpS approach to an organization would be thrown away by its misuse. This isn't as big a problem at CSC as it would be in other organizations, since CSC workers are accustomed to the monitoring of their activities while using OMS, in that accountability for accessing sensitive information about convicted criminals (particularly high-profile offenders) is part of the workplace culture. Nevertheless, even in more controlled environments such as CSC, worker empowerment and wholehearted worker acceptance is critical to the success of PHelpS.

Conclusion

The PHelpS approach has the potential to give real substance to the notion of the "learning organization". By using technology to support human-human collaboration, the spread of knowledge takes place naturally as the informal peer help networks supported by PHelpS distribute new ideas and techniques throughout the organization. Learning is *leveraged* by the technology, not *imposed* through some external agent such as a CAI system. Learning happens authentically in the context of real tasks. Learning happens collaboratively, both through being helped and through helping. In some sense the organization itself can be said to be learning, since the knowledge of how to carry out tasks is rapidly distributed and soon becomes ubiquitous. Interestingly, unlike many information technologies which become bogged down if the number of users grows too big, PHelpS works better the more people who use it, since a large database of potential peer helpers allows a finer grained selection of appropriate ready, willing, and able helpers.

Acknowledgements

We would like to thank the management and staff of the Prairie Regional Psychiatric Centre of the Correctional Services of Canada for their active support and participation in the PHelpS project to date. We would also like to acknowledge the Canadian TeleLearning Network of Centres of Excellence for financial support of the project.

References

Carroll, J.M. (1991) *The Nurnberg Funnel*. The MIT Press.

Chan, T., Chou, C. (1995) *Simulating a learning companion in reciprocal tutoring systems*. In Proceedings of the International Conference on Computer Support for Collaborative Learning. http://www-cscl95.indiana.edu/cscl95/chan.html.

Collins, J.A., Greer, J.E., Kumar, V.S., McCalla, G.I., Meagher, P., Tkatch, R. (1997) *Inspectable user models for just-in-time workplace training.* In Proceedings of the Sixth International Conference on User Modeling, UM97, Chia Laguna, Sardinia, Italy, 327-337.

Constant, D., Sproull, L., Kiesler, S. (1996) *The Kindness of Strangers: The Usefulness of Electronic Weak Ties for Technical Advice.* Organization Science, Vol. 7, No. 2, March-April, 119-135.

Hammer, M. (1996) *Beyond Reengineering. How the process centered organization is changing our work and our lives.* Harper Business.

McCalla, G.I., Greer, J.E., Kumar, V.S., Meagher, P., Collins, J.A., Tkatch, R., Parkinson, B. (1997) *A Peer Help System for Workplace Training.* In Proceedings of the 8th World Conference on Artificial Intelligence in Education, AI-ED 97, 19-22 August, Kobe, Japan, 183-190.

McGraw, K.L. (1994) *Performance Support Systems: Integrating AI, Hypermedia and CBT to Enhance User Performance.* Artificial Intelligence in Education, 5(1):3-26.

Paiva, A., Self, J., Hartley, R. (1994) *On the dynamics of learner models.* In Proceedings of the 11th European Conference on Artificial Intelligence, 163-167.

Palthepu, S., Greer, J.E., McCalla, G.I. (1991) *Learning by Teaching.* In Proceedings of the International Conference on the Learning Sciences, L.Birnbaum Editor, 357-363.

Shepherd, A. (1995) *Task Analysis*. Monk, A.F., Gilbert, G.N. Editors, Perspectives on HCI: Diverse Approaches, London: Academic Press, 145-174.

Slavin, R.E. (1990) *Cooperative Learning: Theory, Research and Practice.* Prentice Hall, Englewood Cliffs, NJ: Prentice Hall.

Wenger, E. (1996) *Communities of Practice: The Social Nature of Learning.* Healthcare Forum Journal, July/August, 20-26.

Authors' Address

ARIES Laboratory, Department of Computer Science, 57 Campus Drive, University of Saskatchewan, Saskatoon, Saskatchewan S7N 5A9, CANADA, e-mail: greer@cs.usask.ca