The *Awareness Monitor:* A Coordination Tool for Asynchronous, Distributed Work Teams

Jonathan J. Cadiz, Susan R. Fussell, Robert E. Kraut,
F. Javier Lerch, and William L. Scherlis
Human-Computer Interaction Institute
Carnegie Mellon University
Pittsburgh, PA USA
+1 412 268 4003
kraut+@andrew.cmu.edu; sfussell+@andrew.cmu.edu

ABSTRACT
Members of interdependent work groups must coordinate their efforts in intricate ways, and they are more successful if they can keep aware of the state of their team, its tasks, and its environment. A major design goal for tools for distributed workgroups is to keep them apprised of important changes without distracting them from their focal tasks. In this paper we describe a system we have developed, the *Awareness Monitor*, designed to provide passive awareness. We discuss key features for passive awareness systems and how they are realized in our system. We also describe the underlying technology and methods used to implement and test the system in a real world environment.

Keywords
Work groups, coordination, situational awareness, passive awareness, cognitive overload, computer-mediated communication, awareness devices

INTRODUCTION
Members of managerial task forces, software engineering teams, flight crews, and other work groups comprised of interdependent roles must coordinate their efforts in intricate ways to be successful. There is substantial evidence that if they can keep aware of the state of the team, task, and environment they will be more successful (e.g., [4, 20]).

To achieve this awareness, teams often rely heavily on active communication to inform each other about status and plans. They conduct face-to-face meetings, talk to each other in hallway conversations, exchange email and distribute status reports to tell each other what is happening.

Team members can also rely on "passive awareness" of the work and team environments to maintain awareness. Passive awareness involves keeping track of events of interest without making a conscious effort to do so. For example, in some settings team members can assess the competence of a new recruits and correct their errors by overhearing conversations [15]. When groups are co-located, getting information requires little effort, and members can maintain passive awareness simply by monitoring activities going on around them.

The task of passively monitoring information about the task, the team, or its environment is substantially more difficult for distributed work groups and requires some degree of technological support. A long-standing goal in both the information retrieval community and the Computer Supported Cooperative Work (CSCW) literature is to develop tools that provide passive awareness. Automated techniques for the selective distribution of information have existed for at least a quarter of a century [21] and are designed to match a changing document flow with a subscriber's interests. Both research [19] and commercial filtering systems for electronic mail apply analogous techniques to private and group correspondence.

Another research stream attempts to build tools to provide collaborators with knowledge of other team members' activity [2]. Many of the CSCW tools for passive awareness have focused on using images and video to provide a view into the remote work environments of other team members (e.g., [1, 8, 12, 14]). Other researchers have attempted to provide information about other people's use of shared documents [13].

Awareness and overload
Limitations on human attention are a major constraint on tools designed to provide passive awareness for distributed work groups. The major problem is that the information needed to maintain awareness of team, task, and environment may overwhelm team members and deflect them from actually doing work. It is difficult, for example, to craft a document if one is continually checking on a teammate's progress. The effective design

of tools to provide passive awareness for work teams must answer two questions:

1. What information is most valuable to provide to the team and what can be ignored?

2. How should one present the information so that relevant information is available if needed, without distracting people from their focal tasks?

What information is it useful for a team to monitor?

The social science literature suggests that collaborative groups would be helped if they could be kept aware of group process information, which is internal to the team, and task environment information, which is external to the team.

Within-team information

The relevant within-team information is likely to be similar across different groups. The following list, while not comprehensive, identifies some important information that seems to lead to group success. Knowledge about work progress and work load allows group members and managers to monitor and assign tasks and to pace their own work [16]. Knowledge about the flow of communication and other indicators of social relationships is helpful in managing the politics of a group and making decisions when values differ (e.g., [5, 17]). Knowledge about who knows what in a group helps members solve problems where distribution of knowledge is the issue [18, 24]. Most awareness tools developed in the CSCW community have attempted to provide team-internal information about work progress and load.

External information

The relevant external information is likely to vary with the typical tasks that a group has to perform [22]. Few CSCW tools have attempted to monitor these environmental events, even though empirical research with teams suggests that this external information is very important. Indeed, maintaining awareness about environmental changes may be more important than maintaining awareness about the team's internal processes; changes in the expectations of a team's "customers", those who make use of the team's product, seem to be especially important. Failure in managerial decision-making teams [7], software engineering teams [6], and new product development teams [3] seems to result when teams ignore changes in their environment.

How should passive awareness information be presented?

The goal of an awareness tool is to help teams monitor changes to important resources while imposing minimally on their attention. Balancing informativeness and intrusiveness is a difficult design challenge [10, 14]. While work on information visualization attempts to display a large amount of information in a form that users can use (e.g., [23]), minimizing attention demands is rarely a design criterion.

There has been no systematic research we are aware of identifying design principles for passive awareness displays. We hypothesize, however, that successful displays will have the following properties, along with others not yet identified:

Proportionality

Displays need to be constructed so that larger or important changes in the environment register larger changes in a user interface.

Asynchronous presentation

By providing information asynchronously one may reduce attentional demands without reducing the usefulness of the information. Because the receivers can fit asynchronous messages into their task schedules, an increase in volume of asynchronous messages leads to substantially less overload than an comparable increase in synchronous messages.

Aggregation

A display that slowly changes to summarize a larger number of individual changes is likely to be less intrusive than one that presents the individual changes directly. For example, rather than presenting all messages in a user's electronic mailbox, the databases used in several organizational memory systems consolidate all messages on a particular topic and provide an indicator to potential users of the volume of messages.

Decomposition

Users will need a mechanism to view the individual changes whose importance has been displayed at aggregate level; that is, they need to be able to move smoothly between aggregate and individual data sources.

Customizability

Because team members have different roles and tasks within a group, awareness tools must allow people to keep aware of different information with different weights. Furthermore, because the value that users put on information will change with time, they need a mechanism both to explicitly indicate their current preferences and to indicate how the importance of the information will change over time.

Dampening

The display should give users a method to acknowledge signals for their attention. Awareness devices can be like alarm clocks that ring if important changes have occurred. Thus, users should have an way to turn off the "alarms" so that their attention is no longer drawn to them.

THE AWARENESS MONITOR

We developed a system called the *Awareness Monitor* to implement and evaluate the design principles listed above. The Awareness Monitor was designed to provide passive awareness of others' activities under the assumption that this will increase coordination and decrease cognitive overload. Although the system can be modified for other

contexts, the external information provided by the Awareness Monitor is specific to a particular user group.

User population

The instantiation of the Awareness Monitor we use to illustrate our design principles was built to increase coordination among teams of students participating in a realistic business simulation called the *Management Game* ("Game"). In this simulation, teams consisting of five to six members compete with one another over the course of a semester. During the simulation, teams make decisions regarding the nature, production, distribution, and financing of their products (watches). They also write reports, make presentations to their boards of directors, trade shares of their own and other teams' companies in a simulated stock market, and deal with crises such as lawsuits and threats of work stoppage. The Game environment is constantly changing: each team's decisions affect the other teams. Team success depends in large part on how well members can coordinate and integrate their activities in this fluctuating environment.

Informational content of the Awareness Monitor

To determine what information the Awareness Monitor should include, we collected data using two methods: first, we conducted a quantitative field study of the Game in a previous year, which included the use of a simple awareness tool. Second, we conducted several interviews with two potential users of the Awareness Monitor.

Previous Game study

Although Game teams have needs for both within-team and external information, we found in a previous study using the same user population that team members are significantly more interested in tools that will help them identify important events in the external environment than they are in tools that provide team-related information [11]. In this study, Game teams were provided with a precursor to the Awareness Monitor, which we called the activity monitoring tool, shown in Figure 1. This system was comprised of a Java applet that organized its information in a tree hierarchy and updated itself every fifteen minutes. With the exception of expanding or collapsing sections of the tree, users could not configure any aspect of the activity monitoring tool.

At the end of the Game, students rated the usefulness of the activity monitoring tool. Respondents showed a clear preference for notifications about changes to their teams' finances and to the Game environment over information about others' availability or changes to shared documents and files.

In the same study, we also surveyed both users and nonusers of the system and asked them to rate how useful it would have been if their team had been automatically notified about new shared documents, changes to existing documents, the availability of team members, changes to the financial condition of their firm, and changes to the

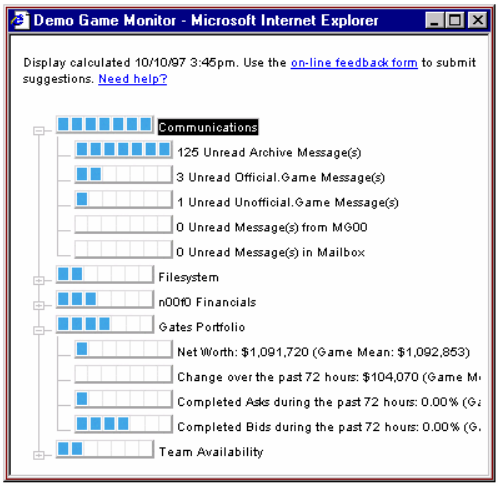

Figure 1: The precursor to the Awareness Monitor we called the activity monitoring tool.

business environment that could affect their firm. Consistent with ratings of the activity monitoring tool, there was greater interest in our developing future awareness tools for financial and Game environment information than for member availability and shared documents (see Figure 2).

We interpreted these findings as indicating that Game teams are satisfied with the manner in which they currently stay aware of one another's activities, document changes, and location (within-team information). A casual examination of team members' email content revealed that a large proportion contained notifications about changed documents and team member availability. However, the requisite external information rarely appeared in email and had to be sought actively.

In light of these findings, the Awareness Monitor was designed to provide a extensive range of external

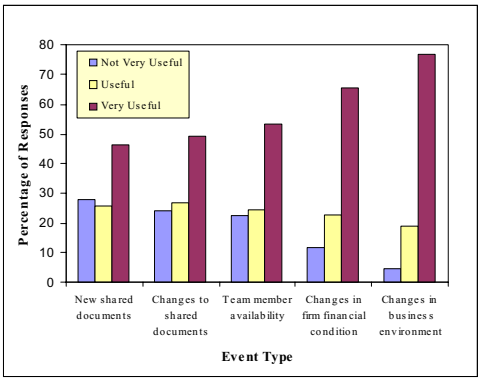

Figure 2: Rated usefulness of automatic notifications by event type.

information along with limited within-team information.

Interviews with users

Two students from Game were hired to participate in several interviews about their team's experiences with Game and their specific informational needs. In the first round of interviews, the students were asked questions about how their teams interacted with the Game's Microsoft Excel spreadsheets. (In Game, most of the major assignments must be completed using spreadsheet templates. Furthermore, considerable amounts of information about the Game world are provided to the students using spreadsheets. Because the Game is a dynamic simulation, these spreadsheets typically change twice a week.)

In the second round of interviews, the students were asked to participate in think aloud usability studies [9]. In the subsequent rounds of interviews, students were asked to help design the default settings for the Awareness Monitor. Results from these usability studies and interviews will be discussed after the Awareness Monitor is described below.

Specific information provided by the Awareness Monitor

Both the data from the previous Game study and the interviews with the Game students indicated that the Awareness Monitor should include the following information:

- Changes to files in the teams' shared directories
- The prices of firms in the stock market
- The amount of shares traded for each firm in the stock market
- The net worth of the user
- The status of the user's proposals to sell or buy stock ("asks" and "bids")
- The values of cells on Excel spreadsheets

THE AWARENESS MONITOR USER INTERFACE

With answers to the questions of what information the Awareness Monitor should present and how it should be presented, we will now describe the Awareness Monitor.

Figure 3 shows the main window of the Awareness Monitor. The basic unit is a *monitor*, which represents a combination of a piece of data (for example, a teammate, the price of a company's stock , or an Excel spreadsheet) and a set of rules to specify how that data should be watched. In the display, each monitor has a title (for example, "net worth watch") and a bar graph icon. The bar graph icon displays between zero and seven bars and is the system's recommendation for the amount of attention the user should pay to a particular information source.

Implementation of the Design Criteria

Below we show how the Awareness Monitor implements the design principles of proportionality, aggregation, decomposition, customizability, and dampening.

Proportionality

The Awareness Monitor was constructed so that more important changes in the environment registered larger changes in the user interface. For example, a monitor that watches a firm's stock market trading volume will have a number of bars proportional to the amount of shares that have been traded for that firm.

Asynchronous presentation

The Awareness Monitor allows for both synchronous and asynchronous presentation. The Awareness Monitor continually changes but is constructed so as not to call attention to itself. The Awareness Monitor can be used asynchronously by hiding the main window (Figure 3) and only checking it when necessary, much like a person may check for e-mail. To minimize attention demand, the Awareness Monitor also provides a small ticker window (Figure 4) which can be placed on the desktop and set to stay visible no matter what the user is doing. The ticker window cycles through the monitors by fading from one item to the next or by scrolling the items across the window, allowing the user to glance up at the window and receive notifications of the current state of data being watched by the Awareness Monitor.

Aggregation

The Awareness Monitor accomplishes aggregation through the use of *monitor groups*. Monitor groups contain collections of monitors and other monitor groups. An example of a monitor group is the item labeled "My

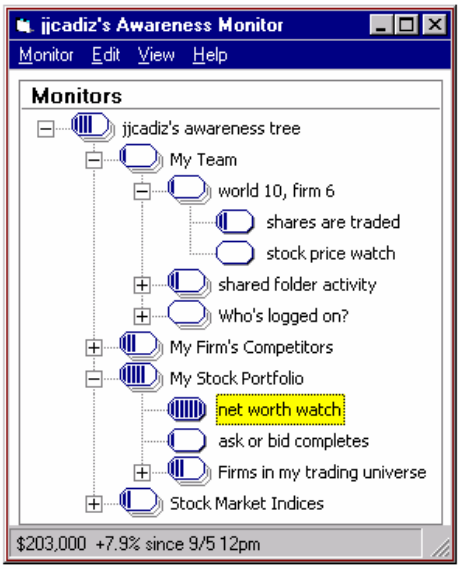

Figure 3: The main window of the Awareness Monitor. The tree is composed of monitors ("stock price watch", "net worth watch") and monitor groups ("My Team", "Stock Market Indices").

Stock Portfolio" in Figure 3. One function of monitor groups is to provide a way to organize monitors; in this regard, monitor groups are similar to folders or directories on a computer's file system.

However, monitor groups serve another powerful function: aggregation. This function is accomplished via the bar graph icon assigned to monitor groups. The number of bars in a monitor group's bar graph icon depends on the number of bars of the items within that monitor group. Thus, a user can determine the amount of attention that should be devoted to an entire set information just by looking at the one bar graph icon for the monitor group.

One crucial component of the Awareness Monitor is the formula used to determine the bar graph icon for monitor groups. The whole point of providing aggregation via monitor groups is to provide summary information to users; however, arriving at the correct formula for summarizing the amount of attention that information should receive, especially heterogeneous information, is difficult. The precursor to the Awareness Monitor (shown in Figure 1) used the maximum formula to aggregate: if one monitor had seven bars and the rest had zero, the monitor group would be assigned seven bars. This aggregation rule does not scale: as more items are aggregated, the likelihood of the group requiring attention increases.

To solve this problem, we used a weighted average to aggregate and determine summary bar graph icons for monitor groups. The weights are assigned by the user using a five-point importance slider bar shown in Figure 5, which will be described in detail below.

Decomposition

Just as users should be provided a way to view summary information, users also need a mechanism to view the individual changes that have contributed to an aggregate display. That is, they need to be able to move smoothly between aggregate and individual data sources.

The Awareness Monitor addresses decomposition in two ways: first, users can expand monitor groups to see the individual items that are contributing to the aggregate bar graph icon; second, the Awareness Monitor provides a piece of specific information for each monitor (for example, the current price of a company's stock or the current value of an Excel cell). When a monitor is selected, this piece of information is shown in the status bar at the bottom of the main window (Figure 3). The specific piece of information attached to each monitor is also displayed on the right-hand side of the ticker window

Figure 4: The ticker window. This window cycles through the monitors by fading from one item to the next or by scrolling items across the window.

(Figure 4), although this information can be hidden to minimize the screen space used by the ticker.

Customizability

Because team members have different roles and tasks within a group, awareness tools must allow people to keep aware of different information using different weights. The Awareness Monitor allows for this type of customizability (see Figure 5). The top half of the rule set dialog box consists of the events to watch. The "events to watch" portion of each rule set is different for each type of data (for example, the possibilities for watching a team member are different than the possibilities for watching a cell on an Excel spreadsheet).

The bottom half of each rule set dialog box is the same for all rule sets: all rule sets are assigned an importance rating using the slider bar in the lower left-hand corner. Users can determine the importance of a rule on a five-point scale. Because the importance of an item often changes over time (for instance, when an item has a deadline associated with it), the lower right-hand portion of the dialog box displays the deadline attached to the monitor, as well as a graph of how the importance of the item changes over time. Clicking the button with the importance graph brings the user to the dialog box shown in Figure 6. This dialog box allows the user to modify the deadline and the way in which the importance changes over time.

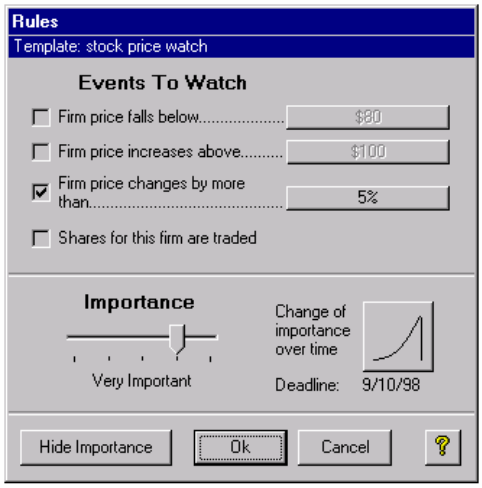

Figure 5: An example of a rule set for watching a firm in the stock market. The bottom portion of all rule sets is the same whereas the top portion differs according to the type of the rule set.

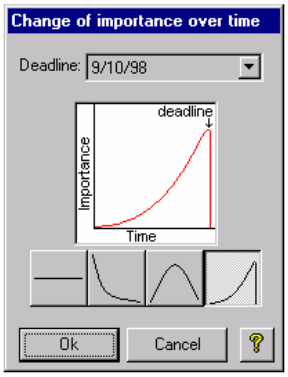

Figure 6: The dialog box used to set the way in which importance changes over time, which may include a deadline.

Two other customizations within the Awareness Monitor are worth noting. First, users can configure the ticker window so that only items with more than a certain number of bars are displayed. This allows users to set thresholds of attention demand, below which the Awareness Monitor will not disturb them. Second, users can modify the window of time over which the Awareness Monitor watches. Some users may want to know about everything that has happened in the past 24 hours whereas other users may want to know everything that has happened in the past week. Possible choices for the window of time include 24 hours, 48 hours, 72 hours, and one week.

Dampening

Users can acknowledge a monitor or monitor group by clicking the item and selecting "reset to zero" under the "edit" menu. Resetting a monitor will cause the Awareness Monitor to ignore all events prior to that moment for that monitor, giving the monitor zero bars. Resetting a monitor group resets all the items within a monitor group.

Configuring the Awareness Monitor

The entire monitors tree (left side of Figure 7) is completely user-configurable. Recall that a monitor consists of a combination of a piece of data to watch and a set of rules describing how that data should be watched. In addition to the main monitors tree, two other trees can be displayed in the main window. The data tree (upper right corner of Figure 7) displays all the things that can be watched by the Awareness Monitor, organized by type. The rule templates tree (lower left corner of Figure 7) contains all the sets of rules that can be used to watch pieces of data. Rule sets are templatized to make creation and modification of the monitors tree as easy as possible.

To give the user a way to determine connections between the items in the three trees, yellow arrows are used. Specifically, when a monitor is selected, the piece of data and rule template attached to that monitor are displayed with yellow arrows next to them. Connections between

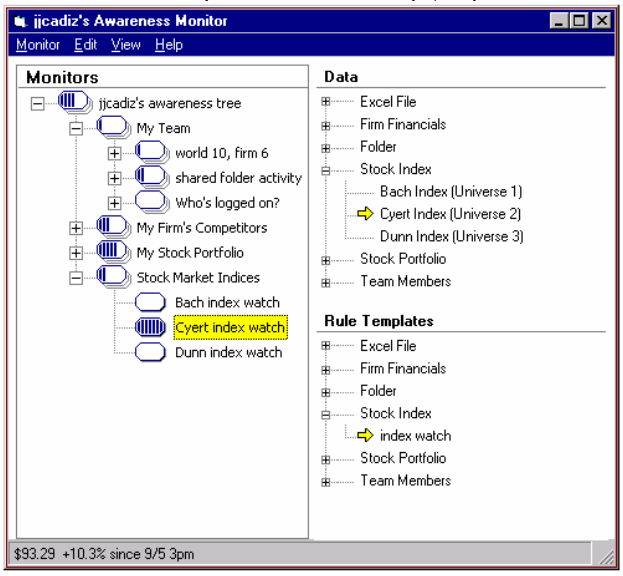

Figure 7: The main window of the Awareness Monitor with the data and rule templates trees shown. The arrows show which items are connected to the selected item.

items in the trees are also shown when a piece of data or rule template is selected.

Defaults provided to Game students

Although it would have been possible for us to ask students to learn the Awareness Monitor and create their own rule templates and monitors, we knew usage would be increased if we minimized the cost of using the Awareness Monitor for the first time. Thus, in our final interviews with the two Game students whom we hired, we determined the best rule templates and monitors tree to give the students.

The default monitors tree is shown in Figures 3 and 7. The four major monitor groups correspond to the four major areas of information that students need to keep aware of.

AWARENESS MONITOR IMPLEMENTATION

Several goals influenced our choice of the technical architecture to use when implementing the Awareness Monitor:

First, all of the information about user settings must kept in a centralized server so that no matter where users log in to the Awareness Monitor, they will see the settings they had from the last time they used the system.

Second, the Awareness Monitor must be able to run over a 28.8 baud modem. All Game students are required to own laptop computers, and many log on to the Internet using a modem to work remotely.

Third, the system must be able to support failures. A system such as this one is bound to fail for any one of many possible reasons. Safeguards against failures are particularly important for a passive monitoring system where a potential exists for users to make decisions based on data that is not recent or no longer accurate.

Using a three tier client/server architecture

To accomplish the first and second goals, the Awareness Monitor system is implemented as a three tier client/server architecture. We describe the system below, starting with the tier that is most removed from the users.

Tier 3: A Microsoft SQL Server database

This database holds all the information necessary for the Awareness Monitor, which directly addresses our first goal. This information consists of two major categories: all the events of interest that occur in the environment (changes to stock prices, files, spreadsheet cells, etc.) and all the settings for each user's Awareness Monitor.

Events resulting from changes in the environment are sent to the database by two programs. The first program only watches changes to the Microsoft Windows NT file system used by the Game students. This program was written in C++ and communicates with an event-based server program implemented using JavaBeans and the remote method invocation (RMI) protocol. This JavaBeans server interfaces with the database using the Java Database Connectivity (JDBC) protocol.

The second program that sends events to the database concentrates on the rest of the events of interest in the Game environment. This program, written using Microsoft's Visual Basic, periodically polls the main Game database for changes in the environment. Any time a change is detected, an event is written to the Awareness Monitor database.

Tier 2: The server-side stored procedures

Our second goal is directly addressed through the use of server-side stored procedures. Programs interact with the database via stored procedures located within the database. The stored procedures are written in Transact-SQL, an extension of SQL that gives SQL much more programmability. Stored procedures create a significant performance boost when running programs over the Internet.

Tier 1: The awareness server & the awareness client

Two main programs work together to interface with the database and deliver awareness information to the user.

The first program is the awareness server. The job of the awareness server is to examine each user's settings and calculate the bar graph icons and specific pieces of information for all the monitors and monitor groups for each user. The awareness server was written using Microsoft's Visual Basic.

The second program is the awareness client, which has the job of interacting with the user. (The awareness client is the program described in the previous sections of this paper.) The awareness client has two major tasks: presenting the awareness information, and interacting with users to allow them to configure their settings. The awareness client is implemented as a web-based ActiveX component and was written using Visual Basic.

Planning for failure

To address our third goal, the awareness client program keeps track of the last time the information for each monitor was updated. If any information is not updated within an hour, a yellow exclamation mark appears in the lower right-hand corner of the main window (Figure 3) along with the time that the monitor was last updated.

EVALUATING THE AWARENESS MONITOR

Two methods are being used to evaluate the Awareness Monitor. First, we are currently conducting a quantitative field study of the Awareness Monitor using teams playing the Management Game. Second, four think aloud usability studies were conducted prior to the release of the Awareness Monitor. From the think aloud studies, we found three major design issues:

First, the interface does not clearly communicate the idea that a monitor is a combination of a piece of data and a rule template. The yellow connection arrows (shown in Figure 7) do not accomplish their intended task. The use of a tree structure for the monitors was good, but using the same structure for the rule templates and data was not. Furthermore, because the connections between the three trees was not obvious, users had a difficult time understanding the use of templates with rule sets.

Second, the five-point importance rating that all rule sets have (Figure 5) is too much for novice users to understand. For this reason, the importance settings were moved to the bottom of the dialog window and are initially hidden; the "show importance" button must be pressed to view them.

Third, the idea that rule templates and pieces of data have a particular type, and the idea that data and rule templates must be of the same type to be combined to form monitors, is not clear. This issue is compounded by the fact that the dialog window used to edit rule templates (Figure 5) looks too similar for different types (only the "events to watch" section changes).

FUTURE DIRECTIONS

The Awareness Monitor is being developed using an iterative design cycle. The first cycle resulted in the tool shown in Figure 1, and the second cycle resulted in the system that is the subject of this paper. For the next cycle, one of our major goals is to address the design issues highlighted by the think aloud usability studies.

Our other major goal is to determine the usefulness of the Awareness Monitor. We are in the process of gathering data from our field study of the Management Game students. These data will allow us to compare Awareness Monitor usage to various measures of team performance. We must answer the question of whether the passive awareness is worth the cost required to maintain it.

REFERENCES

1. Abel, M. J. (1990). Experiences in an exploratory distributed organization. In J. Galegher & R. Kraut

(Eds.), *Intellectual teamwork: Social and technological foundations of group work*. pp. 489-510) Hillsdale, NJ: Erlbaum.

2. Ackerman, M. S., & Starr, B. (1996). Social Activity Indicators for Groupware. *IEEE Computer*, 29(6), 37-44.
3. Ancona, D. G. and D. F. Caldwell (1992), Bridging the Boundary: External Activity and Performance in Organizational Teams. *Administrative Science Quarterly, 37*, 634-665.
4. Cannon-Bowers, J. A., Salas, E., & Converse, S. A. (1993). Shared mental models in expert decision-making teams. In N. J. Castellan, Jr. (Ed.), *Current issues in individual and group decision making* (pp. 221-246). Hillsdale, NJ: Erlbaum.
5. Carley, K. M. & Krackhardt, D. (1996). Cognitive inconsistencies and non-symmetric friendship. *Social Networks*, *18*, 1-27.
6. Curtis, B., Krasner, H., & Iscoe, N. A field study of the software design process for large systems. *Communications of the ACM 31*(11), 1988, 1268-1287.
7. D'Aveni , R. & MacMillan, I. (1990). Crisis and the content of managerial communications: A study of the focus of attention of top managers in surviving and failing firms. *Administrative Science Quarterly, 35*, 634-657.
8. Dourish, P. and Bly, S. Portholes: Supporting awareness in a distributed work group. In *Proc. of CHI'92* (Monterey, California), ACM Press, 1992.
9. Ericsson, K. A., & Simon, H. A. (1980). Verbal reports as data. *Psychological Review*, 87(3), 215-251.
10. Fish, R. S., Kraut, R. E., Root, R. W., Rice, R. (1993). Evaluating video as a technology for informal communication. *Communications of the ACM*, 36(1), 48-61
11. Fussell, S. R., Kraut, R. E., Lerch, F. J., Scherlis, W. S., McNally, M., & Cadiz, JJ. *CSCW'98* (1998)
12. Gaver, W., Moran, T., MacLean, A., Lovstrand, L., Dourish, P., Carter, K. and Buxton, W. Realizing a video environment: EuroPARC's RAVE system. In *Proc. of CHI'92* (Monterey, California), ACM Press, 1992.
13. Hill, W. C. & Hollan, J. D. (1992). Edit wear and read wear. In PO. Bauersfeld, J. Bennett & G. Lynch (Eds.). *Proc. CHI'92 (*pp. 3-9). New York: ACM Press.
14. Hudson, S. E. & Smith, I. (1996). Techniques for addressing fundamental privacy and disruption tradeoffs in awareness support systems. Proceedings, *CSCW'96*. NY: ACM Press.
15. Hutchins, E. 1991. The social organization of distributed cognition. In L. B. Resnick, J. M. Levine, & S. D. Teasley (Eds.), *Perspectives on socially shared cognition* (pp. 283-307). Washington, DC: APA.
16. Kiesler, S., Wholey, D., & Carley, K. (1994). Coordination as linkage: The case of software development teams. In D. H. Harris (ed.), *Organizational linkages*: *Understanding the productivity paradox*. Washington, D. C.: National Academy Press.
17. Krackhardt, David. (1994). The strength of strong ties: The importance of *Philos* in organizations. In N. Nohria & R. Eccles (Eds.). *Networks and Organizations: Structure, Form, and Action*. Boston, MA: Harvard Business School Press.
18. Liang, D. W., Moreland, R., & Argote, L. (1995) Groups versus individual training and group performance: The mediating role of transactive memory. *Personality and Social Psychology Bulletin, 21*, 384-393.
19. Malone, T. W, Grant, K. R., Lai, K., Rao, R. & Rosenblitt, D. (1987) Semi-structured messages are surprisingly useful for computer-supported coordination. *ACM Transactions on Office Information Systems*, *5*, 115-131.
20. Orasanu, J & Salas, E. (1993). Team decision making in complex environments. In G. Klein, J. Orasanu, & R. Calderwood (Eds.), *Decision Making in Action: Models and Methods.* Norwood, NJ: Ablex Publishing Co.
21. Salton, G. *Introduction to Modern Information Retrieval*. McGraw-Hill,1983.
22. Stinchcomb, A. 1990. *Information and Organizations.* Berkeley, CA: University of California Press.
23. Tufte, Edward R. (1990). *Envisioning Information*. New Haven, CT: Graphics Press.
24. Wegner, d.M. (1986). Transactive memory: A contemporary analysis of the group mind. In B. Mullen and G. R. Goethals, (Eds.) *Theories of Group Behavior*. New York: Springer-Verlag.