DT8107 Distributed Information Systems

"Awareness and its support in groupware applications"

Anita Gupta
The Norwegian University of Science and Technology
26.11.2004

Abstract

The goal of this essay is to give an overview of what awareness is and its support in groupware applications. This is a very interesting field, and it could be studied from many different perspectives. Here I focus first on giving definition of awareness and groupware and then categorizing awareness. Afterwards I present my awareness focus which is on workspace awareness and how it is supported in groupware applications such as BSCW and TeamWave. Throughout this essay I will first present an example of how a software development team is using workspace awareness and artifacts for collaboration. After that I will present how a software development team can use workspace awareness provided in groupware applications (BSCW and TeamWave) to collaborate and coordinate their activities. Finally, subject for future research is mentioned.

1. Introduction

Awareness has recently begun to receive considerable attention in CSCW (Computer Supported Cooperative Work) and groupware research [Gutwin and Greenberg, 2002]. While staying aware of others is what we seems to take for granted in the everyday world, maintaining this awareness has proven to be difficult in real-time distributed systems where information resources are poor and interaction mechanisms are foreign. Due to this, working together through a groupware application often seems inefficient and clumsy compared with face-to-face work. It is becoming more and more apparent that being able to stay aware of others plays an important role in the fluidity and naturalness of collaboration. The possibility of supporting awareness of others is looked on as one way of reducing the characteristic awkwardness of remote collaboration. Awareness is a design concept that holds promise for significantly improving the usability of groupware application.

2. Awareness

2.1 Characteristics of awareness

Previous researchers have defined awareness as knowledge created through interaction between an agent and its environment—in simple terms, “knowing what is going on” [Endsley, 1995 p. 36]. This conception of awareness involves states of knowledge as well as dynamic processes of perception and action. Four basic characteristics run through prior work on awareness [Adams, Tenney and Pew, 1995] [Norman, 1993][Endsley, 1995].

1. Awareness is up-to-the-minute knowledge about the state of an environment bounded in time and space.

2. Environments change over time, so awareness is knowledge that must be maintained and kept up to date.

3. People interact with and explore the environment, and the maintenance of awareness is accomplished through this interaction.

4. Awareness is a secondary goal in the task—that is, the overall goal is not simply to maintain awareness but to complete some task in the environment.

Everyone has experienced this kind of awareness; at its most basic, it is what allows us to walk around without bumping into things. As situations and environments become more complex, however, information demands sometimes outstrip our ability to attend, and awareness becomes more noticeable. In these contexts, previous researchers have explored what they call *situation awareness*, a concept that underlies the idea of workspace awareness in groupware.

Adams, Tenney and Pew (1995) and Endsley (1995) are categorizing situation awareness (SA) as:

- Knowledge of a complex system and how it will behave e.g. airplane, mechanical device, surgery.
- Large amount of information and complex combinations of behaviour.
- Little time to react e.g. important to have "good SA".
- Consists of a "**product**" (a piece of machinery, or other environment, and its current state) and a "**process**" (perception of relevant elements of the environment, comprehension of those elements and prediction of the states of those elements in the near future).

Gutwin and Greenberg (2002) are categorizing workspace awareness (WA) as:

- Limited to workspace that: Are flat, medium-sized surfaces, where objects can be displayed.
- Limited to tasks that: View and manipulate artifacts, i.e. generation and execution activities [McGrath and Hollingshead, 1994].
- Limited to groups that: Are medium-sized, 2-5 people, and are involved in mixed-focus collaboration.
- Special case of situation awareness

One aspect of awareness which has begun to generate an increasing amount of interest, concerns the way in which individuals remain sensitive to changes within the immediate environment (workspace awareness) [Heath, Svensson, Hindmarsh, Luff and Lehn, 2002]. The local environment has particular relevance, since it often contains, amongst others things, a diverse and shifting display of different forms of information which are more or less relevant to the activities in which participants engage. It is some analytic and practical interest, how individuals remain sensitive to such a complex array of information and the extent to which they are able to discriminate relevant phenomena and events.

My focus in this essay is on software development team and how the members of such a team can use workspace awareness to collaborate and coordinate their activities and artifacts. The following is an example (the Team Room study) of how a software development team is using workspace awareness and artifacts to collaborate and coordinate their activities.

2.2 Team Room study

A study conducted by Olson, Teasley, Covi and Olson (2002) examined a team that used team room for software development. A team room is a room in which people who are engaged in the same project reside as their primary work site. Team room seems to be more common for projects where there is a high speed for coordination: architects, research teams, designers of a family of appliances, and software engineers. So, in this study the software development team had to produce a software product that was scoped to be completed in six weeks. The team consisted of six people, and the members were diverse in both gender and country of origin and were mixed in whether they had worked together before. All the team members resided in the team room; they did not have other offices to go to or other work responsibilities outside the project. The team room were the size of a large conference room with arrays of desks with workstations, a centre worktable and chairs, and whiteboards and flip charts arrayed around the room. The software development team and their artifacts were in the same room for duration of the project.

The study conducted that the team room in use by the software development team supported interactive, continuous communication. The close quarters of the team room supported impromptu communication and allowed the team members to overhear each other. Subgroups could form and re-form. Over time, this physical reality produced a number of advantages:

- Team members found it easy to develop common ground.
- Team members could engage in standard procedures or methods without having to talk explicitly about them.
- Team members knew each other, and therefore they had transactional memory and expectations about each other's talents, working style, and moods.
- Team members could read each other moment by moment (for example whether someone was having trouble or was in deep thought while coding).
- Team members could use various aspects of the context to assess what was going on at a moment – the position, sound, and gestures of the team-mate.
- Team members had the ability to move between various subactivities of work – some requiring individual work and some group interaction, see Figure 1.
- When team members had questions, often the person who could answer it was at hand.

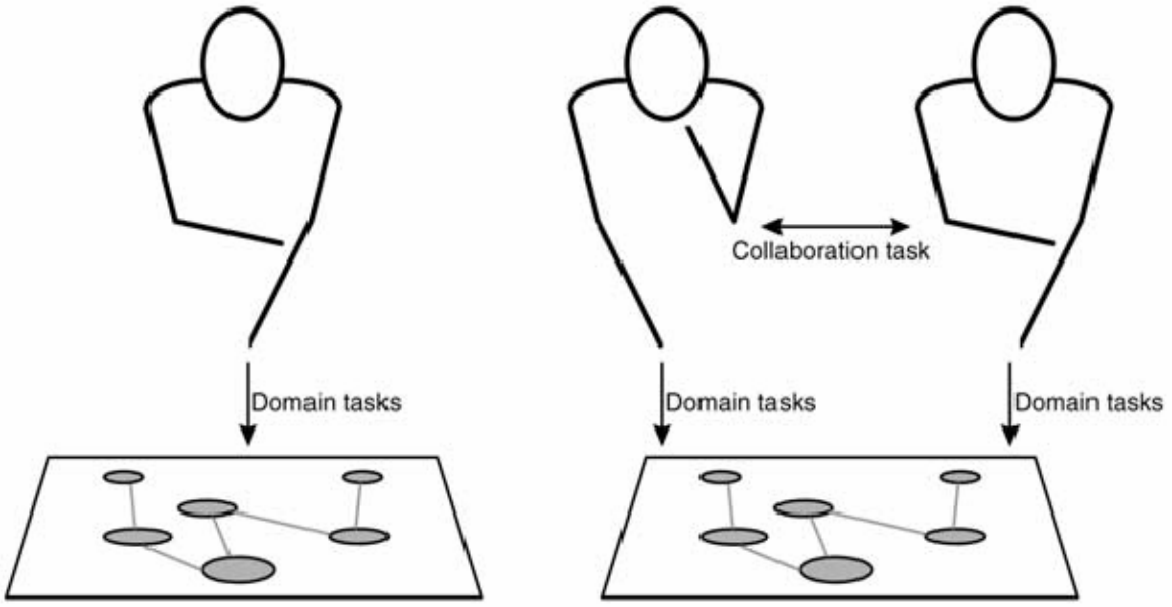

Figure 1: Individual work vs. Collaboration work [Gutwin and Greenberg, 2002]

While being collocated, each member had local control over what he or she was attending to at each moment. The team members were paying attention to other members or artifacts to suit their current goals.

The members of this software development team lived among their artifacts. The team produced a large number of flip charts in the early part of their work (the artifacts represented use cases, object hierarchies, to-do lists and so forth). The artifacts generated by the team members were placed for everyone to see and use. As the team worked, they referred to the posted lists and diagrams often, occasionally moving the artifacts so that comparisons could be made and marking on them to reflect mutually agreed-on changes.

However, collaboration with other people is as likely to take place over distance or time as it is face-to-face [Kraut, Fussel, Brennan and Siegel, 2002]. An abundance of new communication technologies has been developed to mediate remote collaboration: bulletin boards, e-mail, document sharing, videoconferencing, instant messaging, awareness services, and others. The following section is about how workspace awareness is supported in groupware application.

3. Groupware applications

3.1 Groupware

In the eighties the computers and the software were supposed to improve your *own* personal efficiency and the quality of your work. In the nineties, on the other hand the software was supposed to improve the *working groups'* efficiency. In contrast to the most of the computer applications which are designed to support individual work, a groupware application is designed to help people, and specially *groups* of people, to collaborate over time and place. This implies

for example that a working group doesn't need to sit together face-to-face or use the telephone to collaborate. Communication and the distribution of information are done over network of computers. Groupware is a multi-user program that makes it possible for the members of a distributed group to work together by:

- providing group members with communication facilities,
- letting them share their files and data and
- making them aware of each others' existence.

Groupware is the *product* (program) resulting from the research done in the CSCW (Computer Supported Cooperative Work) field, and are focusing on:

- computer systems and
- the design of the computer systems.

There exists several definitions of groupware [Ehrlich, 1999], and a common thing for all of these definitions are that it is difficult to define the concept precise. One definition of groupware that has been common to use is: "*applications that enhance communication, collaboration, and coordination among groups of people.*" [Lotus]

Today's groupware are starting to offer relatively good functionalities to support an effective collaboration. This is due to the research done in the CSCW field. Electronic mail is a widespread success history, and use of videoconference is growing more and more. Shared calendar program was rejected most of the time in their introduction years, but in the modern time this type of applications are becoming more and more accepted then previously.

Workspace systems in general are a special type of CSCW which try to provide a shared workspace for people that are geographically distributed. In this essay I will focus on this type of systems and how they support workspace awareness.

3.2 The awareness problem in groupware applications

In a face-to-face work, awareness of one another is relatively easy to maintain, and the mechanics of collaboration are natural, spontaneous, and unforced. Unfortunately, workspace awareness is much harder to maintain in groupware applications than in face-to-face environments [Gutwin and Greenberg, 2002], and it is often difficult or impossible to determine who else is in the workspace, where they are working, and what they are doing. This can make it difficult to coordinate tasks between the participants. For example, in collaboration at a distance, communication is typically less frequent, characterized by longer lags between messages, and more effortful.

So, how can groupware designers address the workspace awareness problem? Part of the solution is to provide people with more information about their collaborators. As it is infeasible to replicate the detail and size of real-world workspaces designers must carefully determine what information is most important, and how it can be put to best advantage in the application. The first step involves setting out more precisely what workspace awareness is, and the process by which people manage to maintain it.

3.3 Workspace awareness in groupware applications

Gutwin and Greenberg (2002) define workspace awareness as the up-to-the-moment understanding of another person's interaction with the shared workspace. This definition bounds the concept in two ways:

- Workspace awareness is awareness of people and how they interact with the workspace, rather than just awareness of the workspace itself.
- Workspace awareness is limited to events happening in the workspace – inside the temporal and physical bounds of the task that the group is carrying out.

The following is a short description of the three-part framework that examines the concept of workspace awareness and that helps designers understand the purpose of designing awareness support in groupware applications (for a more comprehensive description see [Gutwin and Greenberg, 2002]).

3.3.1 Framework Part one: What information makes up workspace awareness?

Framework part one indicates that workspace awareness is made up of many kinds of knowledge, and therefore it is important to provide designers with a basic idea of what information to capture and distribute in a groupware application. Due to this, it is important to answer "who, what, where, when, and how" questions [Gutwin and Greenberg, 2002]. That is, when we work with others in a physical shared workspace, we know who we are working with, where they are working, what they are doing, when various events happen, and how those events occur. People keep track of these things in all kinds of collaborative work, and these are the kinds of information that should be considered first by designers.

3.3.2 Framework Part two: How is workspace awareness information gathered?

The groupware designer must attempt to present awareness information in ways that make the maintenance of workspace awareness simple and straightforward. This might be easier if people can gather information in familiar ways, even though the actual interface devices in a groupware application may not be familiar. Framework part two is based on understanding the mechanisms people use to gather workspace awareness information from the workspace environment – basically, how people find the answers to the who, what, where, when, and how questions. People obtain this information that is produced by people's bodies in the workspace, from workspace artifacts, and from conversation and gestures (verbal communication) [Segal, 1994][Norman, 1993].

3.3.3 Framework Part three: How is workspace awareness used in collaboration?

The designers of groupware applications need to know the situation and activities where workspace awareness will be used, to better analyze collaborative tasks and to better determine when groupware support is called for. Workspace awareness is used for many things in collaboration, e.g. reduce effort, increase efficiency, and reduce errors for the activities of collaboration [Gutwin and Greenberg, 2002]. According to framework part three there are five

kinds of collaborative activity that are aided by workspace awareness; these are management of coupling, simplification of communication, coordination of action, anticipation and assistance.

Groupware designers can use the three-part framework in two ways:

- As an analysis tool to help them determine the degree of awareness support that is needed for a particular work situation.
- As a guide to determine where in the groupware application interface that awareness support should be provided.

The following are examples of how the three-part framework for workspace awareness supported in the two groupware applications, BSCW and TeamWave can be used to analyze workspace awareness.

3.4 BSCW

The BSCW (Basic Support for Cooperative Work) Shared Workspace System is a Web based groupware application developed at GMD (made publicly available in October 1995) that is used by more than 100,000 users' world-wide [Appelt, 2001]. BSCW is concerned with the integration of collaboration services with existing environments, supporting widely-dispersed working groups with different computing, network and software infrastructures. The software is available for download so that interested organisations may set up their own BSCW servers.

The BSCW application is based on the notion of a "shared workspace" which the members of a group establish for organising and coordinating their work [Bentley, Horstmann and Trevor, 1997]. A shared workspace like BSCW is a repository for shared information, accessible to group members using a simple user name and password scheme. The BSCW shared workspace application is the basis for the services, which include features for uploading documents, version management, group administration and more, all accessible from different platforms using unmodified Web browsers.

3.4.1 Framework Part one: What information makes up workspace awareness?

BSCW provides event facilities for maintaining awareness of other team members' activities in the workspace [Appelt, 1999]. Events are triggered whenever a team member performs an action in a workspace such as uploading a new document, downloading ("reading") an existing document, renaming a document etc. The application records the events and presents these events to each team member in various forms, but the most common way of informing users about events is through the event icons attached to objects. So by using event icons a software development team can obtain information of the team members' activities within the workspace. BSCW distinguishes five types of events:

- New events: indicate that an object has been created since the team member last caught up.
- Read events: show that an object has been downloaded or read by one of the team members.

- Change events: indicate that an object has been modified. This category includes several event types, such as edited, renamed, etc.
- Move events: show that an object has changed its location. This category includes delete and undelete events and cut and drop events
- Touch events: are displayed for a container such as a folder to show that something has happened to an object contained inside (either directly or lower down in the folder hierarchy).

BSCW provides an address book where a team member may collect the names of other team members, and the calendar which contains the dates of all meeting objects related to the team member. Hence, BSCW provide some facilities so that a software development team can answer the "who, what, where, when, and how" questions related mainly to the past.

3.4.2 Framework Part two: How is workspace awareness information gathered?

BSCW gathers workspace awareness information mainly in terms of artifacts and events icons. Artifacts such as UML diagrams, codes, requirement documents, to-do lists etc. can be shared by the team members, and they are visible for all of the team members using the BSCW. Shared artifacts for example, a to-do list can be helpful in coordinating the individual team member's contribution, and make tasks connected to area of responsibility and status (e.g. when a document is started, completed etc.) visible. Artifacts in BSCW can serve as a planning, status and/or evaluation tool, used together with other members or individually. The sorting and searching facilities in BSCW allow team members to specify queries to find artifacts based on names, content or specific properties such as document author or document modification date.
Team members may also add notes (meta-information) to artifacts in the workspace and rate the quality of artifacts, e.g. of documents or URLs that have been created. Event icons can provide information of the team members' activities within the workspace whenever an artifact is exposed to a change, see chapter 3.4.1. Hence, BSCW provide some facilities for how a software development team can find answers to the "who, what, where, when, and how" questions.

3.4.3 Framework Part three: How is workspace awareness used in collaboration?

A software development team can use workspace awareness provided in BSCW to collaborate with other team members. For example, a team member may upload a document he/she has been working on into a workspace. Another team member who is suppose to work further on this document can download it onto his/hers computers, and later upload the same document back into a workspace for review by other team members. Team members may also start a discussion on any topic they like and the application presents the threads in a style similar to the Internet newsgroups, and the team members can distribute documents in a BSCW workspace to specified team members via email. There is also possible for team members to specify synchronous sessions and launch respective tools, e.g. audio/video conferencing software or shared whiteboard applications. BSCW allows team members to have a meeting, by creating a so-called meeting objects which are particularly useful for the preparation of meetings since they include features such as selection of participants, automatic invitation of participants who may accept or decline an invitation, or the distribution of meeting notifications via email. Documents within BSCW workspace can be put under version control which is particular useful for joint document production, or they may be locked during an editing session to prevent other team members from

accessing documents temporarily. Hence, BSCW provide some facilities so that a software development team can collaborate and coordinate their activities and artifacts.

Although the BSCW application primarily supports asynchronous modes of communication, it also provides some features for synchronous collaboration by using synchronous communication tools such as chat or audio/video conferencing.

3.4.4 BSCW supporting software development activities

Information in BSCW is provided in terms of artifacts and event notifications and it supports certain types of activities in software development, for example:

- Awareness information (most asynchronous awareness)
- Communication facilities
- Configuration management
- Change management
- Coordinate tasks, artifacts etc.
- Create information and/or artifacts
- Share information and/or artifacts
- Categorize artifacts
- Start a discussion
- Make decisions
- Plan and perform meetings
- Transition between asynchronous and synchronous collaboration

3.5 TeamWave

TeamWave Workplace is a commercial Internet groupware product based on a room metaphor (e.g. physical rooms with features that imply how people can use them for both individual and collaborative activities, which include how people naturally share expertise). TeamWave was originally based on a research prototype called TeamRooms, developed at the University of Calgary [Greenberg and Roseman, 2003].

3.5.1 Framework Part one: What information makes up workspace awareness?

TeamWave provide several facilities for maintaining awareness of other team members in the room [Greenberg and Roseman, 2003]. These facilities provide both a general awareness of who is around and a more fine-grained awareness of others' actions in the room:

- Room Users: Each room displays a list of team members in the current room.
- Telepointers: Within the workspace itself, telepointers – one for each member – communicates gestures to provide a fine-grained sense of awareness of the actions of other team members.
- Room overview radar: Because the room is larger than will fit on a single display, a radar view provides a stylized miniature overview of the entire room.

- Applets: Each applet is a special-purpose groupware application, which team members can include in their rooms. For example, a room used to manage the software project that the team members are working on may have applets for task, lists, bug-reporting forms, a pointer to an online version of the project's specification, etc.
- Group memory: TeamWave keeps a version history of the state of each room and each applet, and that makes it possible for team members to browse and retrieve earlier versions. This allows the team members to compare two versions of a room, for example, to review earlier stages in the software project.

A team member can use the "Logged in Users" window in TeamWave. This window will lists all the software development team members currently connected to the team's server as well as the room each member is currently working in. More information on team members can be found by double-clicking a person's image or by selecting "Info" from the "Logged in Users" window. Hence, TeamWave provide some facilities so that a software development team can answer the "who, what, where, when, and how" questions related mainly to the present but also to the past.

3.5.2 Framework Part two: How is workspace awareness information gathered?

TeamWave gathers workspace awareness information in terms of room users, room overview radar, applets, group memory and artifacts (I have given a description of the five first facilities in chapter 3.5.1). In TeamWave the same artifacts are used for asynchronous and synchronous collaboration, and they can be used either by a single team member or by several of the team members at the same time. Team members can use the shared whiteboard (occupies the "wall" of each room) for producing freehand drawings (e.g. UML diagrams, codes etc.) and add text. Artifacts such as UML diagrams, codes, requirement documents, architecture documents, to-do lists etc. can be shared by the team members, and they are visible for all of the team members using the TeamWave. Shared artifacts for example, a to-do list can be helpful in coordinating the individual team member's contribution, and make tasks connected to area of responsibility and status (e.g. if a document is started, completed etc.) visible. Artifacts in TeamWave can serve as a planning, status and/or evaluation tool, used together with other members or individually. Hence, TeamWave provide some facilities for how a software development team can find answers to the "who, what, where, when, and how" questions.

3.5.3 Framework Part three: How is workspace awareness used in collaboration?

A software development team can use workspace awareness provided in TeamWave to collaborate with other team members. For example, a team member can create new room by choosing a command from the Room menu, specifying a name for the room. Each room can then be customized to suit the specific need of the software development group and their tasks (e.g. one room for activities associated with requirement specification phase and another for design phase). Team members can send a message to another logged-in member, even if they are in different rooms, by selecting "Page" from the "Logged in Users" window. TeamWave also provides a simple text-based chat tool is provided that allows team members to type messages to each other when they are in the same room. A software development team can conduct a formal meeting, since rooms can be configured with appropriate tools and information ahead of time, by having all tools support real-time interaction, and by making artifacts in a room persist over these meeting phases. The team members can smoothly shift from individual to group work, since the

application makes no technical distinctions between single-user and group rooms, or between single-user and groupware applications and artifacts contained by a room. When team members are present in a room at the same time, they are automatically working together synchronously, and when the team members work in the rooms at different times, they can work asynchronously just by leaving things in the room. Hence, TeamWave provide some facilities so that a software development team can collaborate and coordinate their activities and artifacts.

Although the TeamWave application primarily supports synchronous modes of communication, it also provides some features for asynchronous collaboration such as leaving things in the room when team members work at different time, as well as sending messages to one another with e-mail, and to all the other team members with e.g. bulletin boards.

3.5.4 TeamWave supporting software development activities

Information in TeamWave is provided in terms of room users, telepointers, room overview radar, applets, group memory and artifacts and it supports certain types of activities in software development, for example:

- Awareness information (most synchronous awareness, but also some aspect of asynchronous awareness)
- Communication facilities
- Change management
- Coordinate tasks, artifacts etc.
- Create information and/or artifacts
- Share information and/or artifacts
- Start a conversation/discussion
- Make decisions
- Plan and perform meetings
- Transition between asynchronous and synchronous collaboration
- Transition between individual and group work

3.6 Towards unified groupware applications

We can by section 3.4 and 3.5 about BSCW and TeamWave see that both of the groupware applications provide various tools for asynchronous/synchronous collaboration, they provide a transition from individual to collaborative work and provide a set of awareness functionalities. Even though the two groupware applications that I have analyzed provide awareness information they leave it to the users all the overload of the coordination of cooperative activities. This leads towards the concepts of coordinative protocols and awareness in unified groupware applications, which will help the users to coordinate their cooperative activities and will provide asynchronous and synchronous collaboration equally.

Unified groupware applications have been achieved through the design and implementation of a series of prototypes [Schmidt and Simone, 2000]:

- *Coordinative protocols and artefacts:* A language (Ariadne) has been developed for defining, specifying, and executing "coordinative mechanisms"; this language is implemented in an agent-based architecture (ABACO) that offers a specialized Interoperability Language to support flexibility at various levels of depth [Schmidt and Simone, 2000].
- *Mutual awareness:* A set of linguistic features for the definition of mutual awareness both between actors and artificial entities have been identified. These features, which have been implemented in a software component called AW-Manager [Schmidt and Simone, 2000].
- *Interoperability:* A software component (called Reconciler) supports the interoperability of different "coordination mechanisms" in terms of the mutual alignment of their boundary objects and events [Schmidt and Simone, 2000].

These prototypes are built on a mandatory requirement; both the protocols and awareness should be functionalities adapted by the users in their current situation.

4. Conclusion

The goal of this essay was to give an introduction to awareness and its support in groupware applications. Awareness can be studied from many different perspectives, so it was impossible to cover all aspects of the subject. My focus, however, was on the workspace awareness. An interesting problem not explored here, due to the limited size of this essay is how awareness can support e.g. XP programming.

Radical collocation affords a great deal of ease in coordinating the work among team members, but collaboration with other people is as likely to take place over distance or time as it is face-to-face. Therefore, the design of technology that meets the requirement of integrating different aspects of awareness, while taking into account the distributedness of cooperative work as well as the semantic level of awareness, not only requires suggestive metaphors and adequate models but also languages with the appropriate expressiveness and compositionality. It is only then users are provided with a set of possibilities for tailoring the technology according to their dynamic needs. Although the designed prototypes have been conceived in the light of the desired integration, the achievements are presently more at architectural level than at the functionality level that can be effectively and flexible used.

A subject for future research might be to construct a comprehensive and coherent technological framework for the problem of presenting contextual information of activities in complex work settings, especially in the case of awareness information, which becomes extremely crucial in view of the current user interfaces with respect to such demanding presentation problems.

5. References

[Adams, Tenney and Pew, 1995] Adams, M., Tenney, Y. and Pew, R.; *Situation Awareness and the Cognitive Management of Complex Systems* in: Human Factors, vol. 37, no. 1, 1995 pp. 85–104.

[Appelt, 2001] Appelt, Wolfgang; W*hat Groupware Functionality Do Users Really Use? Analysis of the Usage of the BSCW System*, 2001, pp. 1-5. (http://bscw.fit.fraunhofer.de/Papers/PDP2001/PDP2001.pdf)

[Appelt, 1999] Appelt, Wolgang; *WWW Based Collaboration with the BSCW System*, 1999, pp. 1-13. (http://bscw.fit.fraunhofer.de/Papers/SOFSEM99/sofsem.pdf)

[Bentley, Horstmann and Trevor, 1997] Bentley, Richard, Horstmann, Thilo and Trevor, Jonathan; *The World Wide Web as enabling technology for CSCW: The case of BSCW* in: Computer Supported Cooperative Work (CSCW), 2-3, 1997, Kluwer Academic Press, Amsterdam, pp. 111-134. (http://bscw.fit.fraunhofer.de/Papers/CSCWJ-WWW/CSCW_journal.html)

[Endsley, 1995] Endsley, M.; *Toward a Theory of Situation Awareness in Dynamic Systems* in: *Human Factors*, vol. 37, no. 1, 1995, pp. 32–64.

[Ehrlich, 1999] Ehrlich, K.; *Designing Groupware applications: A Work-Centered Design Approach* in: Computer Supported Co-operative Work, M. Beaudouin-Lafon (ed.), John Wiley & Sons, Chichester, UK, 1999, pp. 1-28.

[Gutwin and Greenberg, 2002] Gutwin, C and Greenberg, S.; *A Descriptive Framework of Workspace Awareness for Real-Time Groupware* in: Computer Supported Cooperative Work (CSCW), (under publishing), 2002, pp. 47.

[Greenberg and Roseman, 2003] Greenberg, Saul and Roseman, Mark; *Using a Room Metaphor to Ease Transitions in Groupware* in: Ackerman, M.S, Volkmar, Pipek and Volker, Wulf: Sharing Expertise, Beyond Knowledge Management. Cambridge, MIT Press, 2003, chapter 9. ISBN: 0-262-01195-6.

[Heath, Svensson, Hindmarsh, Luff and Lehn, 2002] Heath, Christian, Svensson, M.S, Hindmarsh, Jon, Luff, Paul and vom Lehn, D.; *Configuring Awareness* in: Computer Supported Cooperative Work (CSCW), vol 11, issue 3, 2002, pp 317-347.

[Kraut, Fussell, Brennan and Siegel, 2002] Kraut, R.E, Fussell, S.R, Brennan, S.E. and Siegel, J.; *Understanding Effects of Proximity on Collaboration: Implications for Technologies to Support Remote Collaborative Work* in: P. Hinds and S. Kiesler (eds.): Distributed Work. Cambridge, MA: MIT Press, 2002, pp. 137-162.

[Lotus] - The technical resource for Lotus software; www.notes.net

[Norman, 1993] Norman, D.; *Things That Make Us Smart*. Reading, 1993, MA: Addison-Wesley.

[McGrath & Hollingshead, 1994] McGrath, Joseph E. & Hollingshead, Andrea B.; *Groups Interacting With Technology*, Sage Library of Social Research 194, Sage Publications Inc, 1994. ISBN: 0-8039-4898-0.

[Olson, Teasley, Covi and Olson, 2002] Olson, J.S, Teasley, S., Covi, L. and Olson, G.; *The (Currently) Unique Advantages of Collacted Work* in: P. Hinds and S. Kiesler (eds.): Distributed Work. Cambridge, MA: MIT Press, 2002, pp. 113-135.

[Schmidt and Simone, 2000] Schmidt, Carl and Simone, Carla; *Mind the gap! Towards a unified view of CSCW* in: COOP2000, Sophia Antipolis, France, 23-26 May, 2000.
[Segal, 1994] Segal, L.; *Effects of Checklist Interface on Non-Verbal Crew Communications*, NASA Ames Research Center, 1994, Contractor Report 177639.