Using a Room Metaphor to Ease Transitions in Groupware

Saul Greenberg and Mark Roseman

Research Report 98/611/02

Department of Computer Science, University of Calgary
Calgary, Alberta, Canada T2N 1N4
phone: +1 403 220 6087
email: saul@cpsc.ucalgary.ca *or* roseman@teamwave.com

Cite as:

Greenberg S. and Roseman, M. (1998). Using a Room Metaphor to Ease Transitions in Groupware. Research report 98/611/02, Department of Computer Science, University of Calgary, Calgary, Alberta, Canada, January.

Using a Room Metaphor to Ease Transitions in Groupware

Saul Greenberg[1]
Department of Computer Science
University of Calgary
Calgary, Alberta CANADA T2N 1N4
Phone: +1 403 220 6087
Email: saul@cpsc.ucalgary.ca

Mark Roseman[2]
TeamWave Software Ltd.
100 Discovery Place One
3553 31 Street NW
Calgary, Alberta CANADA T2L 2K7
roseman@teamwave.com

ABSTRACT

Many groupware systems contain gaps that hinder or block natural social interaction or that do not let people easily move between different styles of work. We believe that the adoption of a room metaphor can ease people's transitions across these gaps, allowing them to work together more naturally. Using the TeamWave Workplace system as an example, we show how particular gaps are removed. First, we ease a person's transition between single user and groupware applications by making rooms suitable for both individual and group activity. Second, people can move fluidly between asynchronous and synchronous work because room artifacts persist. People can leave messages, documents and annotations for others, or work on them together when occupying the room at the same time. Third, we ease the difficulty of initiating real time work by providing people with awareness of others who may be available for real-time interactions, and by automatically establishing connections as users enter a common room. Fourth, we discuss how a technical space can be transformed into a social place by describing how a group crafts meaning into a room.

Categories and Subject Descriptors

C.2.4:Distributed systems/distributed applications;
D.2.2 Tools and Techniques/user interfaces;
D.2.10 Design Methodologies: Representation;
H.4.3 Communications Applications/Computer conferencing and teleconferencing;
H.5.1 Multimedia Information Systems/Artificial realities;
H.5.3 Group and Organization Interfaces/Asynchronous interaction/Synchronous interaction/Theory and models

Keywords

Spaces, places, rooms, groupware, computer supported cooperative work, human computer interaction.

---

[1] Saul Greenberg leads the GroupLab project, which developed the research foundations leading to TeamWave.
[2] Mark Roseman is the chief designer and architect of TeamWave Workplace.

1. Introduction

1.1. Gaps in groupware

In 1987, DeSanctis and Gallupe proposed a typology of group decision support systems that indicates how particular technologies bridge time and space. This typology was then applied to groupware by Johansen (1988) and subsequently propagated in almost every introduction to CSCW (e.g., Nunamaker et al 1991; Shneiderman 1992; Dix, Finlay, Abowd and Beale 1993; Baecker 1993; Preece 1994; Baecker, Grudin, Buxton and Greenberg 1995). Table 1, for example, shows a typical space/time matrix and its four quadrants.

	Same Time	**Different Time**
Same Place	*Face to face interactions* • conference tables with embedded computers • public displays • dedicated tools for e.g., voting and brainstorming	*Ongoing tasks* • team rooms • group displays • shift work groupware • project management
Different Places	*Distributed real time interactions* • chat systems • transparent sharing of single user applications • collaboration-aware groupware • video conferencing • media spaces	*Communication and coordination* • unstructured or semi-structured electronic mail • electronic bulletin boards • asynchronous conferencing • list servers • workflow systems • schedulers • collaborative hypertext

Table 1. A typical space/time matrix (after Baecker, Grudin Buxton and Greenberg 1995 p.742)

Various authors have extended this typology to include further dimensions (Baecker, Grudin, Buxton and Greenberg 1995) such as:

- differences in group size (DeSanctis and Gallupe 1987);
- whether there are multiple individual *versus* group sites (Nunamaker et al 1991);
- whether the time and place are predictable or unpredictable to participants (Grudin 1994);
- informal *versus* formal encounters (Preece 1994);
- refinement of same time/different times to distinguish access to a system as concurrent synchronized, serial, mixed, and unsynchronized, as well as to distinguish the "granularity" of sharing in terms of the object chunk size and the frequency of update (Dix, Finlay, Abowd and Beale 1993).

The problem is that these taxonomies, while useful for classifying CSCW issues and technologies, do not reflect the ways people work together in practice. In the real world, people move continually and effortlessly between different styles of collaboration: across time, across place, across formality, and so on. Unfortunately, many groupware technologies were designed to handle only the limited collaborative activity or situation indicated by a single cell within the matrix, as can be seen by the examples in Table 1. Consequently, when people move between styles of collaboration, they must switch from one groupware application to another. This introduces a barrier or *gap* that interferes with people's normal collaborative patterns. More formally, gaps are defined as the physical or perceptual boundaries within groupware that either distract participants from the work they are doing, or that block them from crossing the spatial, temporal, or functional boundaries inherent in collaborative work[3] (Ishii, Kobayashi, and Grudin 1993). To move across these gaps, people now make fairly heavy-weight and disruptive transitions within and between software. Alternatively, they may decide that the personal cost is too great and do without groupware support.

There are many other gaps in groupware, and some of those identified in the literature are raised here[4]:

- the gap between individual and shared work, where people have difficulty moving themselves, their working styles, and their artifacts between a personal working area and the group's working area (Ishii, Kobayashi, and Grudin 1993; Baecker 1993, Chapter 14);
- the technology gap that exists when groups use both conventional software and groupware (Baecker 1993, Chapter 14), and when groups work in heterogeneous computer environments.
- the gap between synchronous and asynchronous work, where either different time or same time interaction is supported, but not both (Baecker 1993, Chapter 14);
- the gap between different phases of a collaborative activity, where people need to move between different work tasks; one example is the movement between pre-, during-, and post-meeting activities, where people move between meeting preparation, the actual meeting, and meeting cleanup (Dubs and Hayne 1992; O'Grady and Greenberg 1994);
- the gap between the desire to collaborate and actually establishing a groupware session (Cockburn and Greenberg 1993);
- the gap between same place and different places, where part of a group that is trying to meet are co-located in a single room, another part in another room, and the rest in their own offices (Nunamaker et al 1991; Baecker 1993, Chapter 14);
- the gap between informal and formal activities (Nunamaker et al 1991; Preece 1994);
- the gap between computer and desktop tools, where physical artifacts and tools (such as paper documents and pencils) cannot inter-operate with electronic artifacts and tools (such as a paint program) (Ishii 1990).

One goal of modern groupware research is *seamlessness,* defined as mitigating or "eliminating unnecessary obstructing perceptual seams" or gaps (Ishii, Kobayashi, and Grudin 1993). Similar views have been expressed by other authors. Some assert that gaps must be bridged if groupware is to be effective (Baecker 1993, Chapter 14), and others have called for any-time, any-place groupware (Baecker, Grudin, Buxton and Greenberg 1995).

In this paper, we argue that a groupware systems supporting a room metaphor can nurture a wide range of collaboration styles within a single environment. As we will report shortly, the affordances of a room metaphor imply functionality that helps mitigate or remove technical gaps, and that supplies users with a conceptual model that reduces perceptual seams. Consequently, the transitions that people make as they move between their styles of collaboration are eased.

1.2. An overview of the paper

We will begin by briefly listing some basic features of the room metaphor and how it affords individual and collaborative work. Throughout the rest of this paper, we will use the TeamWave Workplace system as a case study of a groupware system designed around a room metaphor. Section 3 will introduce its technical features and user interface.

In Sections 4 to 7, we introduce four different gaps, and discuss how the room metaphor as realized in TeamWave Workplace can ease people's transitions across these gaps. In Section 4, we show how rooms are equally suitable for supporting both individual and group work. In particular, the transition between individual and group activity is just a function of the way a room is used and of the number of people in a room. In Section 5, we show how making room artifacts persistent lets people move fluidly between synchronous and asynchronous work. People can leave messages, artifacts and annotations for others, or work on them together if they occupy the same room at the same time. We illustrate this with an example that shows how people move through the pre, during, and post meeting processes of a formal meeting. In Section 6, we show how rooms ease the difficulty of establishing real time connections by providing awareness of who is available for real time interaction, and by automatically making connections when

[3] Gaps, seams, and boundaries are all terms used by various authors to express similar concepts.
[4] A similar list is raised in Baecker (1993) in his chapter on the future of groupware for CSCW.

they enter a common room. In Section 7, we discuss how a technical space can become a social place by the way people craft meaning into particular rooms. The paper closes with a review of related work.

2. Features Inherent in a Room Metaphor

The crafting of physical space and how it can be used by people is well known in the field of urban planning and architecture (e.g., Alexander 1979). Similarly, the crafting of virtual space is being studied in Human Computer Interaction (e.g., Henderson Jr. and Card 1986; Kuhn and Blumental 1996; Harrison and Dourish 1996; Benford, Brown, Reynard and Greenhalgh 1996). In this section, we extend that work to develop an (incomplete) list of features suggested by a room metaphor. We concentrate on features that afford both individual and collaborative work, and their implications to the design of electronic rooms.

A room metaphor is a particular type of spatial model whose features afford a natural way to provide people with good collaborative opportunities (e.g., Kuhn and Blumental, 1996). The rooms we consider are analogous to physical rooms used by teams within an organization. Such rooms typically include (Covi, Olson and Rocco 1998):

- personal offices;
- shared rooms available to several teams, where its use may be scheduled and there is an expectation that the room is returned to its former state after use e.g., breakout rooms and conference rooms;
- live-in dedicated project rooms, which (usually) contain open offices housing a team of 3-6 people, and a shared space where people can work together and leave artifacts in it e.g., skunk works and team rooms;
- non live-in dedicated project rooms which are similar to the above except that team members have their offices elsewhere e.g., war rooms;
- public spaces for social interaction and casual work e.g., coffee rooms, foyers, and commons.

The features described in this section and summarized in the first two columns of Table 2 arise from particular properties inherent in physical rooms. Rooms are bounded spaces; they act as containers; items within a room have spatial locations; and people can inhabit a collection of rooms. Later sections will describe how particular features can be represented electronically, as summarized in the rightmost column of Table 2.

2.1. A room as a bounded space

The first property of a room is that it is a *bounded space*, and affords the features of partitioning, containment, and permeability.

Walls can *partition* a large space into a collection of rooms. Because of partitioning, the distance and barriers (walls) between rooms can separate and/or bring people and their artifacts together (Harrison and Dourish 1996). Individual rooms act as *containers* (Kuhn and Blumental 1996). They contain people, furniture, tools for work and communication (e.g., telephones, overhead projectors), documents, and other artifacts that can support both individual and group activities. Finally, rooms are *permeable*. People can enter and leave them (Kuhn and Blumental 1996), and items can be brought into them or can be removed from them. People can also look into a room, perhaps because the door is open or because there is a window in it.

2.2. A room as a container

Second, rooms are containers that afford the features of persistence and customization.

Objects left in a room *persist* over time. Consequently, people can place and store objects in it (Kuhn and Blumental 1996). When people leave a room, they expect its objects to remain undisturbed in the same spot on their return, unless someone else had entered it in the interim. As persistent containers, people can now *customize* a room by bringing in their own artifacts and by manipulating artifacts already within the room. The way rooms are customized often indicates its *ownership* and its *privacy* i.e., who is allowed in them (Kuhn and Blumental 1996). For example, Covi, Olson and Rocco (1998) studied the way cognitive artifacts are used in dedicated project rooms. They noticed that these rooms contain flip charts and whiteboards, as well as items taped to the walls. These act as shared visual displays that display work in progress, current status of tasks, reference materials and so on.

Feature	Description	How it is supported in TeamWave
Bounded space		
Partitioning	Rooms are collections of bounded spaces, separated by walls.	The space is partitioned as a set of discrete rooms, individually presented within a large window. Individual people and objects can be in only one room at a time.
Containment	Rooms can contain people, tools, and artifacts.	Rooms contain generic tools for communication (e.g., chat) and work (e.g., shared whiteboard and the drawings created on it). Rooms also contain any number of applet instances and the artifacts created within them.
Permeability	People can enter and leave rooms, look inside them through doors and windows, and bring things in and out of them.	People can navigate between rooms. Doorways to other rooms can be placed within a room. People can also see who is in other rooms and their levels of activity. However, people cannot see what is going on in a room unless they enter it, and people can only copy items to other rooms.
Container		
Persistence	Objects left in a room persist over time in the same spatial location.	All rooms and their current state persist over time, even when the server is shut down and turned on again. This includes marks on the whiteboard, and applets and their contents. However, for privacy reasons, people can only see the chat text typed while they were in the room.
Custom-ization	People can customize a room by bringing in their own artifacts and arranging them in the space.	People customize a room by creating it, by marking the wall, by adding applets to it, by setting access controls, and by using the applets. Customizations persist over time.
Privacy and ownership	The way rooms are customized often indicates its ownership and who is allowed in them.	Rooms are given names, and have an owner. Access control rights can be set explicitly. Door states can be set to indicate how public or private a room is.
Spatial location		
Spatial relations	Objects can be organized within a room when a person spatially relates them to one another.	Applets in a room can be placed close together, and their positions persist until they are moved. Marks on the whiteboard can also be situated next to applets.
Proximity and action	Collaborators can interpret each other's actions by how close they are to one another and to the objects in the space.	People's position in a room, its whiteboard marks, and its applets and contents are indicated by telepointers and radar overviews.
Common reference and orientation	People see and reference the room, its objects, and its inhabitants from a similar orientation.	A room appears identically to all the people within it. Through scrolling, however, people can have different viewports into the room. Even though people may see different portions of the room, all have a small radar overview that proves a birds-eye view of the entire room.
Reciprocity	Collaborators know that others can see their actions and objects in the same way.	When people are in the same part of a room, all objects and the fine grained actions taken over them appear immediately on all displays.

Table 2. Features of physical rooms and how they are supported in TeamWave Workplace (continued…)

Features	Description	How it is supported in TeamWave
Inhabitation of the space		
Presence and awareness	We sense other's presence and activities as we navigate between rooms and glance into doorways.	A peripheral window shows all the people present in the space, and what room they are in. Another window shows a list of rooms, and what people are in them.
Encounters	We meet and occasionally initiate encounters with people as we navigate between rooms, and as others enter our rooms.	People can send a message to other people seen in the windows mentioned above. Alternatively, they can enter their rooms, which automatically initiates communication channels.
Habitation	Different rooms can be inhabited by one or more people, or be empty.	As with physical rooms, TeamWave rooms can be empty, or populated by any number of people at any time.
Real time meeting definition	A meeting occurs merely by having two or more people in a room.	As with physical rooms, a meeting occurs whenever two or more people are in a room. Communication channels are automatically opened, and people can collaborate over all items within a room.
Asynch-ronous definition	Asynchronous collaboration occurs when people leave things for others in a room.	As with physical rooms, asynchronous collaboration occurs whenever a person leaves a note or artifact in a room. Because items persist, anyone later entering the room can see them.

Table 2 (continued). Features of physical rooms and how they are supported in TeamWave Workplace

2.3. Spatial locations of objects within a room

The third property of rooms is that objects occupy spatial locations within them, which affords the features of spatial relations, proximity and action, common reference and orientation, and reciprocity.

The *spatial relations* between items in a room are indicated by their spatial location relative to one another. Consequently, people can organize the room by associating, aggregating, and ranking its objects (Kuhn and Blumental 1996). For example, Covi, Olson and Rocco (1998) noticed that software developers using flip chart sheets in their dedicated skunk works room moved and clustered the sheets at various times e.g., sheets were placed side by side so that people could look for similarities between the software requirements shown on them. As another example, teams collaborating over a whiteboard or table top often specified relations of text and graphics by clustering them together (Tang 1991). Yet again, Leiva-Lobos, De Michelis and Covarrubis (1997) mentions that "the physical arrangement of the work space makes the historical and spatial context of the project visible to its participants". Next, collaborators use *proximity of actions* to interpret each other's activities by seeing how close people are to one other and to the objects in the space (Harrison and Dourish 1996). Because objects are fixed in a common location, people within the room can have a *common reference and orientation* to one another and to the objects within the space (Harrison and Dourish 1996; Benford, Brown, Reynard and Greenhalgh 1996; Tang 1991). They all see and reference the room, its objects, and its inhabitants from a similar orientation. Similarly, collaborators can expect *reciprocity* as they know that others can see their actions and objects in the same way (Harrison and Dourish 1996). In essence the objects and their arrangement within a room becomes part of a shared visual display (Covi, Olson and Rocco 1998). However, the degree of common reference, orientation and reciprocity around the display depends on where people are directing their attentions (Greenberg and Gutwin 1996)

2.4. Rooms as an inhabited collection

The final property of rooms is that they can be inhabited by people. Combined with other properties, this affords the features of presence and awareness, encounters and the way rooms can be inhabited. It also defines both real time meetings and a limited form of asynchronous interaction.

People inhabiting a collection of rooms show *presence and awareness*. Habitation means that individuals are present in the space, and that others can sense their presence and activities as they move between rooms and as they glance into them. Consequently, people meet and occasionally initiate *encounters* with one another as they navigate between rooms, and as they enter rooms with other people in them (Kraut, Egido, and Galegher 1988; Kuhn and Blumental 1996). Next, different rooms can be *inhabited* by one or more people, or be empty. However, the size and layout of particular rooms can restrict how many people can be reasonably accommodated. The way people use the space also defines different types of interaction. A *real time meeting* can be defined as merely having two or more people inhabit the same room, as they can now communicate and collaborate with one another. In contrast, *asynchronous collaboration* occurs when people leave things for others in a room: they know that the other people will be inhabiting the space sometime in the future.

In summary, physical rooms have features that imply how people can use them for both individual and collaborative activities. While the features may be common sense, realize that most groupware systems do not support the features and corresponding collaboration opportunities suggested here, or do so in an awkward manner. To contrast, we suggest that the reader juxtapose the offerings of typical systems based on the notion of 'groupware as tool' (e.g., a shared whiteboard application) with the offerings of groupware based on a room metaphor. One example of such a room-based system is TeamWave Workplace, which is presented in the next section and used as a case study in the remainder of this paper.

3. TeamWave Workplace

TeamWave Workplace (*http://www.teamwave.com*) is a commercial Internet groupware product based on a room metaphor. TeamWave was originally based on a research prototype called TeamRooms, developed at the University of Calgary. In previous publications, we provided an overview of TeamRooms (Roseman and Greenberg 1996a), implementation details (Roseman 1996; Roseman and Greenberg 1997), and produced a video demonstrating its interface (Roseman and Greenberg 1996b). In turn, TeamRooms was influenced by our earlier work developing the GroupKit groupware toolkit (Roseman and Greenberg 1996c).

The features and user interface of TeamWave Workplace described in this paper reflect an early version of TeamWave (released in early 1997) that was very similar to the TeamRooms prototype. Later versions of TeamWave (detailed at the TeamWave Web site) differ significantly from the version presented here.

In this section, we briefly describe the interface and features of TeamWave Workplace. We first show how a user starts the system, and how they can create a new room or enter an existing room. The standard groupware tools in every room (such as the shared whiteboard) are then described. We then give an overview of the applets, which provide more specific support for a group's activities. We defer examples of how TeamWave can be used in particular situations to subsequent sections. Finally, Table 2 summarizes how the features of a room metaphor, listed in Section 2, are supported in TeamWave.

Figures 1 and 2 illustrate the main components of the TeamWave user interface. The large window on the bottom of Figure 1 shows a user (Carl) sharing a room called "TeamWave Demo" with two other users (Saul and Mark). The bottom window in Figure 2 shows the same room as seen by another user (Saul) who is looking at a different part of the room. The top windows in Figures 1 and 2 show various peripheral windows.

Setup. TeamWave uses a client-server architecture, where a server on the network maintains a set of rooms for a community. Administrators set up the membership of a TeamWave community by specifying what users are allowed to connect to the server. Administrators may also define sub-groups and their membership, which allows an end-user to assign access control rights to a room by specifying a group's name.

Starting Up. Through a traditional login dialog, users connect their client to the TeamWave server that maintains the set of rooms for their community. They are then placed in a default room, and their screen will look similar to the one shown in Figure 1 (excepting the business card window). Before detailing the contents of the rooms window, we first describe how users can navigate between rooms, and create new ones.

Entering other rooms. Figure 1a displays the "Rooms on this Server" window that lists all rooms currently available to the community. A user enters another room by selecting it from this list. Though primarily used for navigation, the list also displays which users are in each room. For example, Adam is in the "Foyer" room, while Saul, Mark, and Carl are in the "TeamWave Demo" room. The door icons are primarily social

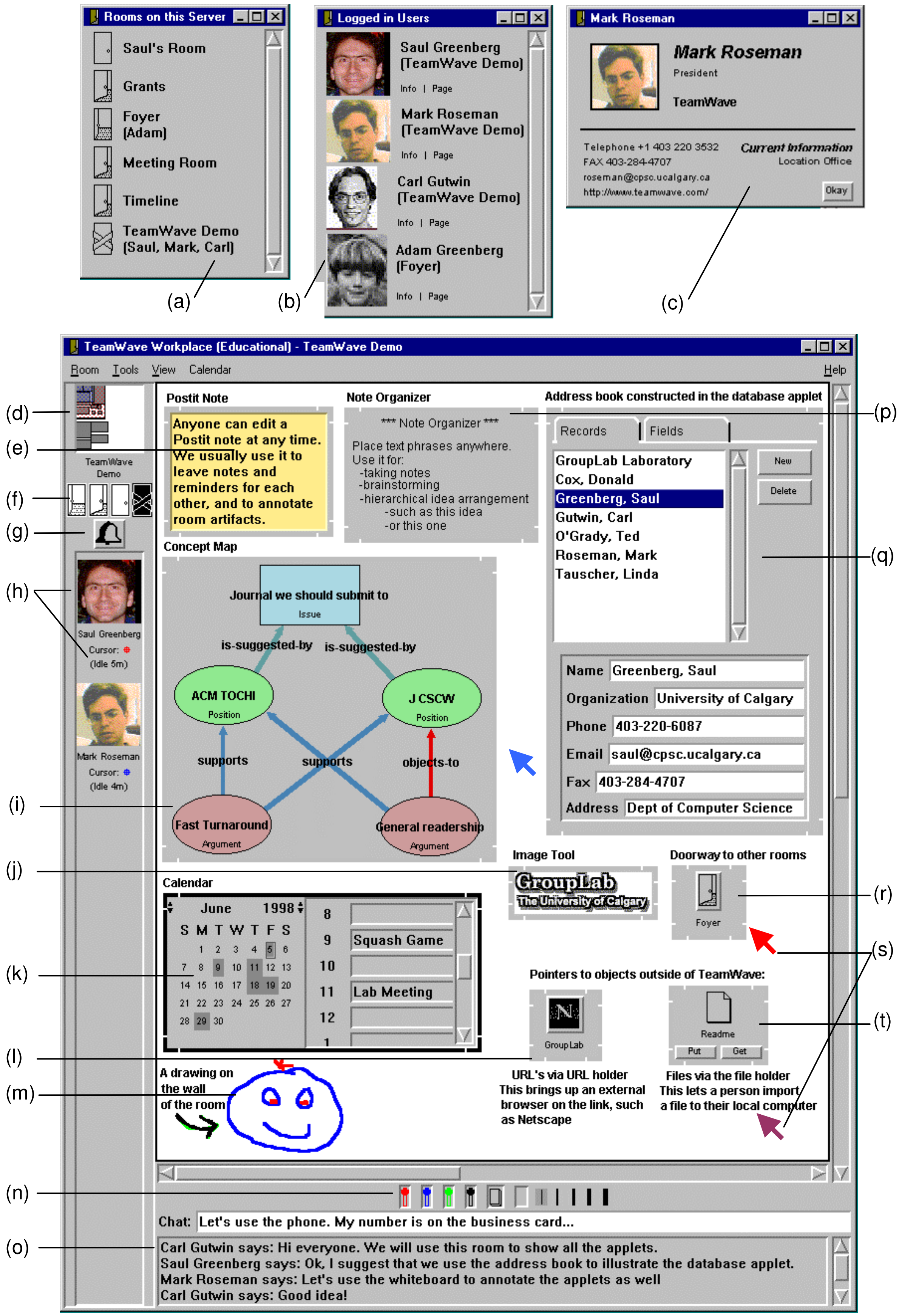


Figure 1. The TeamWave Workplace user interface, showing a room and peripheral windows as seen by user Carl.

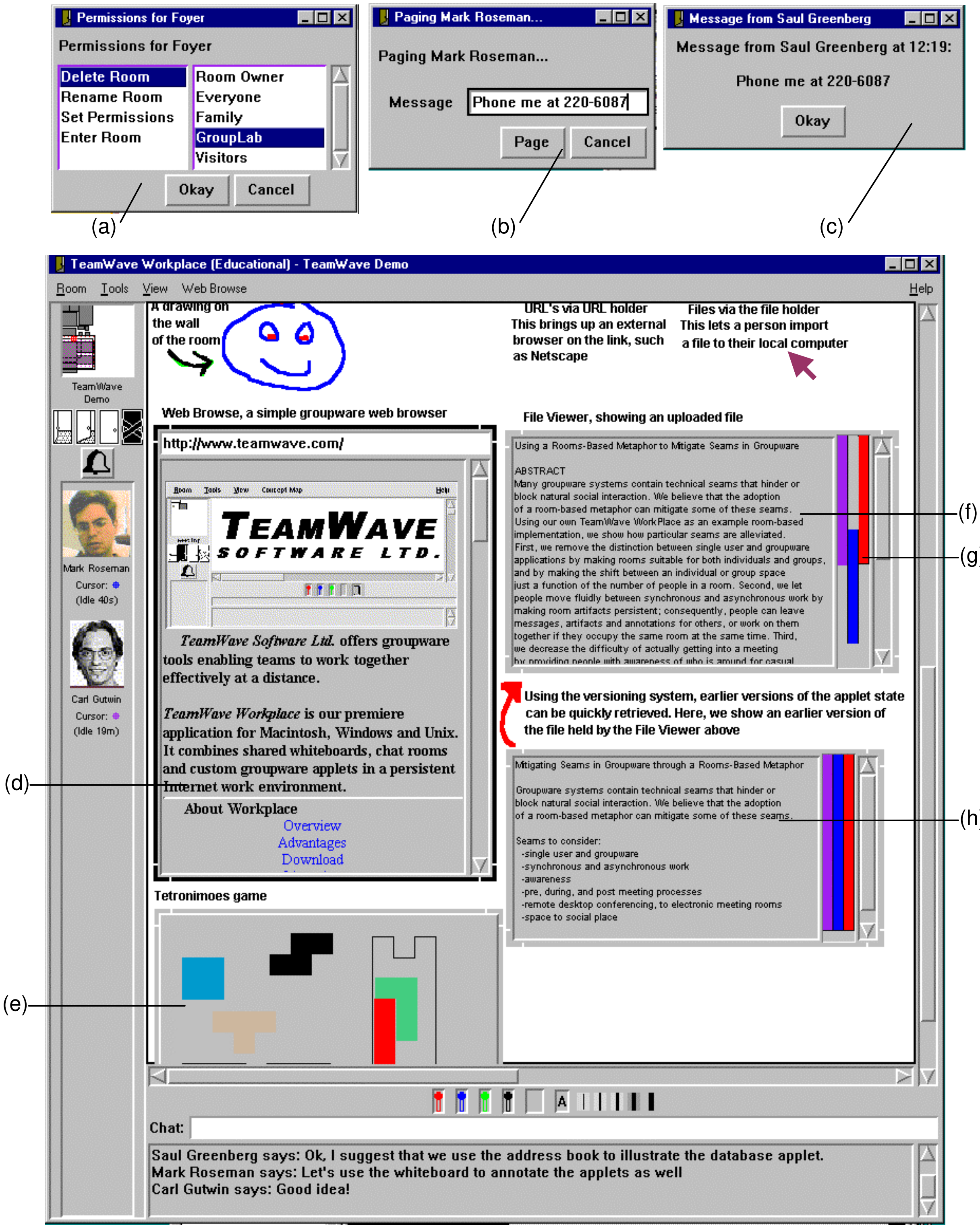


Figure 2. Another user's (Saul) view of the same room, where he has scrolled to a different position within it. Other peripheral windows are also illustrated.

indicators; open doors suggest public rooms, while closed doors suggest private rooms. The icons also reflect system access rights e.g., a locked door prevents users from entering the room.

Creating new rooms. Users create new rooms by choosing a command from the Room menu, specifying a name for the room. By default, anyone in the community can enter, change or even delete the room. If desired, the room's creator can limit these actions to specific subgroups (Figure 2a).

Seeing other users. Users also see a "Logged in Users" window that lists all people currently connected to the group's server, as well as the room each user is currently working in. In Figure 1b, for example, we see that four people are currently logged on. Each person's image is normally a scanned-in picture provided by the user. In one experimental version of TeamWave, the still pictures could be replaced with video snapshots taken several times per minute.

Seeking information about other users. More information on users can be found by double-clicking a person's image or by selecting "Info" from the "Logged in Users" window. This will display a "business card" window that includes phone and fax numbers, email addresses, personal home page URL, and the person's physical location (Figure 1c).

Standard communication features. Though each room in TeamWave can be customized to suit the specific needs of the group and their task, certain generic facilities are provided that are useful for a wide range of groups and activities.

- *Paging.* A user can send a message to another logged in user, even if they are in a different room, by selecting "Page" from the "Logged in Users" window (Figure 1b). This raises a dialog allowing the user to compose a short message (Figure 2b). The message will appear in a pop-up window on the other person's display (Figure 2c).
- *Chat.* A simple text-based chat tool (Figure 1o) is provided that allows users to type messages to each other when they are in the same room. While digital audio or video conferencing can provide an excellent complement to TeamWave, the system does not require such facilities, thereby accommodating users without the necessary hardware or bandwidth.
- *Shared whiteboard.* A shared whiteboard occupies the "wall" of each room. Users select different colored pens or the eraser from the pen tray (Figure 1n), and can produce freehand drawings on it (e.g., Figure 1m). They can also add text to the wall, as shown by the labels in the room.

Awareness Features. TeamWave provides several facilities for maintaining awareness of other team members in the room. These facilities provide both a general awareness of who is around, and a more fine-grained awareness of others' actions in the room (Gutwin and Greenberg 1996).

- *Room users.* Each room displays a list of users in the current room, as shown in Figure 1h. The local user is not shown on the list, which is why the images in Figures 1 and 2 differ. The user list also shows idle times for each user, which indicates how long it has been since that person was active in the room (i.e., since they last typed or moved the mouse).
- *Telepointers.* Within the workspace itself, telepointers—one for each user—communicate gestures to provide a fine-grained sense of awareness of the actions of other users (Figure 1s). Telepointers are colored, and each person can see the color of the other people's telepointers on the user list (Figure 1h).
- *Room overview radar.* Because the room is larger than will fit on a single display, a radar view (Figure 1d) provides a stylized miniature overview of the entire room. The radar shows the locations of all applets in the room, the position of each user's viewport into the room, and miniature telepointers to show the location of their mouse cursor. As users move around the room and manipulate applets, the radar tracks their actions. Because users can scroll independently to different parts of the room (as shown in Figures 2 and 3, where only a small part overlaps), the radar lets each person track where the others are and what they can see.

Applets. While the facilities described previously provide fairly generic support for collaboration, applets in TeamWave are designed for more specific needs of the group. Each applet is in fact a special-purpose groupware application, which team members can include in their rooms as needed by selecting them from the 'Tools' menu. For example, a room used to manage a software project may have applets for task lists, bug reporting forms, a pointer to an online version of the project's specification and so on. In contrast, a 'coffee room' might have applets for a card game, online comics, or electronic postcards left in the room by travelling colleagues. As another example, the particular room illustrated in Figures 1 and 2 were set up to display and annotate all TeamWave applets for a demonstration.

As also shown in Figures 1 and 2, each applet is embedded in its own frame within the room, in a similar fashion to OpenDoc or OLE/ActiveX components (Orfali, Harkey and Edwards 1996). At any time, users may select new applets from the Tools menu, choosing from a list of available groupware applications. Users can move, resize, and delete applets. All such changes are immediately visible to all users in the room. Applets can also add commands to the global menu bar (e.g., there is a Calendar menu label in Figure 1 because the Calendar applet has the focus).

TeamWave supports any of the types of applications that could be constructed in GroupKit (Roseman and Greenberg 1996c), including text editors, drawing tools, card games, meeting tools, and so on. Applets are fully group-aware, allowing shared views, immediate updates of fine-grained actions, and simultaneous editing. A partial list of applets included in TeamWave is provided in Appendix 1 and illustrated in Figures 1 and 2.

Group memory. TeamWave keeps a version history of the state of each room and each applet. Versions are automatically saved when the last user in a room leaves, or when a user explicitly saves a snapshot. Users can browse and retrieve earlier versions. If a user asks for a previous version of a room, the complete state of the room is replaced with the earlier version. If they ask for a previous applet version, a new instance of that applet is created in the room that holds the earlier version. This allows users to compare two versions of an applet, for example, to review earlier stages in a project. We see in Figure 2 that a group has used the File Viewer applet (Figure 2f) to view a paper that was being written by a group member. Through the versioning mechanism, the group retrieved a much earlier version of the paper (Figure 2h), allowing them to compare the differences.

Low common denominator technology. Because groupware must be accessible to the entire group, TeamWave requires only a modest technology infrastructure. First, TeamWave works with low bandwidth networks, including 14.4 modem links. Second, TeamWave runs on most popular systems, including Windows 95/NT, MacOS, and several flavors of Unix. Thus people in a TeamWave community can use quite different platforms yet still collaborate with each other.

We now move from a description of TeamWave Workplace to considering how systems that support a room metaphor (like TeamWave) can ease transitions in groupware.

4. Transitions Between Individual and Group Work

Everyday working styles of people shift regularly and easily between individual and group activity, which implies that software should support both styles. Unfortunately, most systems are categorized either as a single-user application, or as multi-user groupware. By forcing them to switch tools, this introduces a gap for people moving between individual and group activities.

Single-user applications offer little or no explicit support for people who want to work together through computers. While people do often use single user systems collaboratively (Nardi 1991), the effort to do so across time and distance barriers is high. On the other hand, groupware built to fit only group needs is often inadequate for supporting individual work. The gap between these two categories often means that people cannot use the same tools for conceptually similar tasks that cross over individual and group work (Ishii, Kobayashi, and Grudin 1993; Baecker 1993, Chapter 14). People must take on the additional workload of shifting between tools, and learning how to use new tools. They must translate common artifacts, such as documents, into formats amenable to both single and groupware systems.

In this section, we show that the gap between these two categories is unnecessary, and that it is possible to have lightweight transitions between single and group activities. First, we will consider individual work as *reflexive CSCW*, where individual work can be seen as a form of cooperative work. Next, we introduce how the room metaphor can ease transitions between individual and group work, a theme that will be revisited in detail in later sections.

4.1. Considering individual work as cooperative work

Cockburn and Thimbleby (1991) suggest that CSCW environments can be problematic when there is a disparity between systems used for personal work and those used for cooperative work. They argue for a *reflexive perspective* on CSCW that blurs the distinction between personal and group work, claiming that an individual often behaves in ways that resemble a group's behaviour. They suggest that cooperative environments must not only support group activity, but must also cater to individual requirements. If these environments are useful for personal work, they will become familiar and predictable, which in turn minimizes the extra effort usually associated with learning a dedicated groupware tool.

The reflexive perspective of CSCW suggests that individual work has properties of group behaviour, where individuals adopt at least two roles when working. In a *personal management role,* a person coordinates and schedules personal activities for their future selves, such as creating to do lists, leaving reminders for oneself, and so on. Within a *worker role,* the person actually carries out the steps of the activities. When there is more than one project and more than one working environment (such as when the physical work is spread over several machines in different locations), then the person spends much time communicating and coordinating with their 'future selves', almost as if their future selves were actually different people.

4.2. TeamWave and individual work

The room metaphor in TeamWave Workplace has been used in exactly this reflexive manner. For example, consider the personal room shown in Figure 3, which is stylized from an actual room created and used by one of the authors. We see room artifacts created for personal management, such as the "To Do" list, a calendar for personal scheduling, and a note that reminds the room's creator about urgent actions. We also see that the room contains on-going work that they can review and work on whenever they are in the room. These include the concept map of interface methodologies being developed for a course, and a partially completed paper review. Finally, the room contains pointers to objects not visible in this room. These include a file titled 'Teamwave.txt' that can be uploaded and downloaded (which is a paper being written), and a doorway pointing to the room "Grants" that contains documents relevant to current grant applications.

TeamWave implements several features of a room metaphor (Table 2) that makes it a suitable environment for supporting an individual's personal management role and worker role: persistence, customization and ownership; spatial relationships of objects; and machine-independence.

1. The persistent nature of a room and the fact that it can be customized creates a sense of personal space, context and ownership. When its owner enters it, all items are exactly as they had been left. Multiple rooms means that people can create multiple contexts to reflect different tasks, and that personal task switching is quickly accomplished by moving to the appropriate room (Henderson Jr and Card 1986; Card and Henderson Jr 1987; Card, Robertson, and Mackinlay 1991). In contrast, users of traditional graphical environments must re-establish their context by remembering relevant files, opening them, positioning and sizing them on the screen in a meaningful way, and moving to the correct portion of the file.
2. Relationships between personal management activities and work artifacts can be made by spatial proximity and by annotation. As an example, the note reminds the room's owner that both the camera ready copy of the paper and the referee report is due tomorrow. The note's appearance and position on top of the to do list suggests urgency. It is also close to the artifacts being referred to. The paper (teamwave.txt) is next to the note, and the paper review is clearly visible. (Alternatively, the person could have used the whiteboard to draw lines between the note and the items). The positioning of doorways to other rooms next to some relevant information is yet another way to link related items. In contrast, traditional systems do not link personal management tools with work artifacts.
3. TeamWave is not tied to any individual machine, which means that people can access their rooms anywhere, anytime, and from most platforms (e.g., Macs, PCs, Unix). Unlike pages on the World Wide Web (which are also accessible anytime, anyplace), all room artifacts are editable. People can also make non-TeamWave objects accessible within a room by uploading them into the room as a file. For example, a person can import a Microsoft Word file into the room, and then download it onto a different machine later on. In this case, the file is viewed and edited outside of TeamWave.

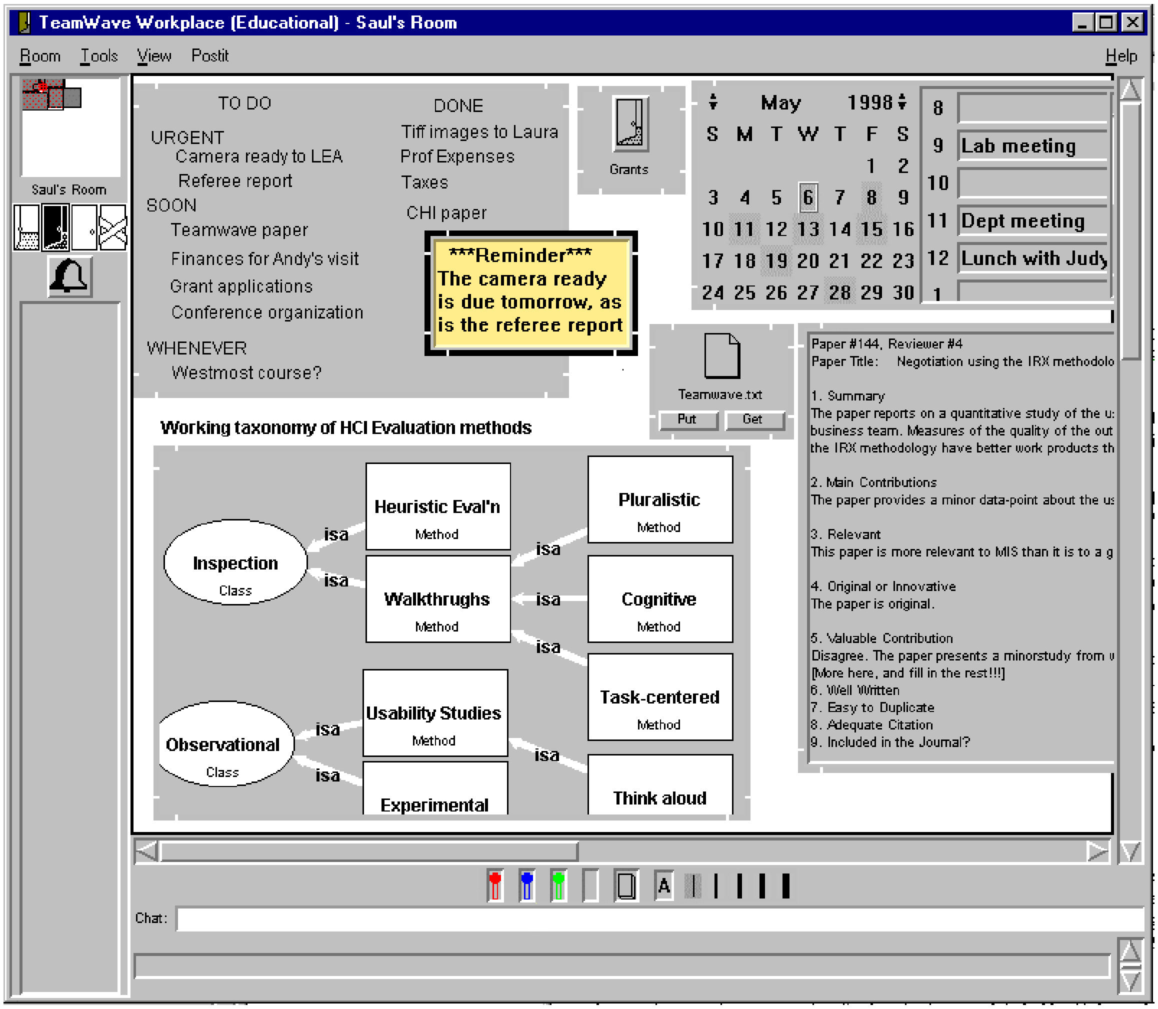


Figure 3. A TeamWave room that supports individual work. Personal management tools include the to do list, the calendar, the reminder, as well as the immediate visibility of ready to hand work artifacts. These artifacts include pointers to other rooms, documents, and on-going work.

In summary, the notion of persistence, customization, ownership and spatial relationships inherent in the room metaphor, as well as the cross-platform capabilities of TeamWave, facilitates the tenants of reflexive CSCW by supporting personal coordination and management, and any-time, any-place access to work artifacts.

4.3. Flowing between individual and shared work

The next step is to see how rooms facilitate people smoothly shifting between individual and group work, a theme that will be explored in detail over subsequent sections.

Physical rooms can be empty, or occupied by a single person, dyads, small groups and (space allowing) large numbers. People can bring items into a room, either for their own individual use, to leave for others, or to work on together with others. For example, Covi, Olson and Rocco (1998) observed that physical rooms afford easy transitions from individual to group work, where team members report that "being co-present during individual work provided opportunities for interruption at a moment's notice for important interactions". The room itself is passive; what determines whether the room (including its contents) is a group or individual space and whether a meeting is occurring (see Table 2: real time meeting definition) is simply the entry and departure of people, and social convention.

The room metaphor in TeamWave Workplace is similar to its physical counterpart. The system makes no technical distinctions between single user and group rooms, or between single user and groupware applications and artifacts contained by a room. We have already seen that a room can provide a good personal management and working space for individuals. The exact same facilities make it a good group management and working space. The same room becomes a group space whenever a second person enters it. If a person enters an already occupied room, the room and its applets behave as fully functional real time groupware, and synchronous collaborative work can be pursued. If a person enters a previously occupied room, then they can see what items have been left by others. That is, the room's persistence can cause it to behave as asynchronous groupware that supports group management activities.

In summary, people can make the transition between individual and shared work smoothly because rooms make no technical distinction between individual and group tools. The same tools for personal management and work become tools for group management and group work. As in real life, it is the individuals and groups that determine how the space is used, rather than the technology.

The following sections will delve into the finer points of this transition between individual and shared work. Section 5 discusses how people move between asynchronous and synchronous group work. Section 6 describes how rooms support casual interaction, which is a common way that people move between individual and group work.

5. Transitions Between Synchronous and Asynchronous Collaboration

Most groupware supports either synchronous or asynchronous collaboration, but not both. In this section, we briefly review the major styles of groupware within these categories, show how a room metaphor lets people move fluidly across the synchronous / asynchronous gap, and illustrate through a scenario a group's blending of asynchronous and synchronous activities over an entire meeting process.

5.1. Groupware for synchronous and asynchronous work

There are several major categories of asynchronous systems (Table 1, right side). People can send messages to one another with electronic mail, and to groups with list servers and bulletin boards (Sproull 1991; Sproull and Kiesler 1991). Group members can carry on lengthy but focused conversations with asynchronous conferencing systems that maintain conversational threads about specific topics (Turoff 1991; Sproull and Kiesler 1991). They can also negotiate and schedule activities on group calendars (Lange 1992). Managers and workers can coordinate commitments through semi-structured and structured messaging systems (e.g., Object Lens by Malone, Lai and Fry 1992; the Coordinator by Flores et.al 1988; Lotus Notes™). Teams can track activities through workflow systems (Abbot and Sarin 1994). Organizations can also post and retrieve documentation comprising an organizational memory through hypertext systems that act as an organizational memory (Conklin 1992).

Similarly, there are different categories of synchronous systems (Table 1, left side). These support real-time communication ranging from textual chat systems, to audio channels, to video phones. Recent advances in communication systems increase the sense of tele-presence, where inter-personal cues are transmitted such as body language, gaze direction (Ishii, Kobayashi and Grudin 1993), gestures (Tang 1991; Greenberg, Gutwin and Roseman 1996), and spatial relations between participants (Buxton 1992). Other systems support real-time collaboration over work artifacts. These include application-sharing (Greenberg 1990), groupware drawing systems (Greenberg, Hayne and Rada 1995), groupware text editors (Baecker, Nastos, Posner and Mawby 1993), live presentation tools, as well as business meeting tools for brainstorming and idea organization (Valacich, Dennis and Nunamaker 1991).

For a group to switch between asynchronous and synchronous interaction, they must switch the tools they are using. This is a heavyweight transition. Applications must be found and started, communication channels must be established, documents created in asynchronous tools (e.g., an email attachment) must be imported into a groupware application that allows them to be shared and edited, and so on.

5.2. How rooms permit asynchronous and synchronous activity

A room metaphor offers a simple way for a group to move between asynchronous and synchronous work. First, when team members are present in a room at the same time, they are automatically working together synchronously. Unlike groupware applications that require people to create and establish separate connections between each tool and communication channel, a room acts as a single connection point. The act of entering a room immediately connects all people within it, both for communication (e.g., the chat tool) and for work (e.g., all applets and the background whiteboard).

Second, when people work in the room at different times, they can work asynchronously just by leaving things in the room. In a study of real project rooms, Covi, Olson and Rocco (1998) observed that teams leave coordination documents for each other: to-do lists, action items, telephone logs, vacation schedules, and so on. TeamWave supports this type of asynchronous activity through the same features that support reflexive CSCW. Persistence allows work artifacts and annotations to be left in a room for others to review or change at a later time. Spatial proximity can link these messages and artifacts. For example, notes, reminders and comments can be left on the whiteboard or written on PostIts next to a relevant work object.

Third, the same artifacts work for both asynchronous and synchronous work. All room objects can be used either by a single user or by several users at the same time. Team members can work on tasks individually, and then share task artifacts with others, either asynchronously or synchronously. The transition between asynchronous and synchronous activity becomes a function of how people use the rooms and its tools, rather than a function of system constraints. Of course, this same capability also helps ease the transition between single user applications and groupware, as described in Section 4.

With these three capabilities, the room metaphor as implemented in TeamWave supplies the basic framework for easing the asynchronous / synchronous transition. New applets or tools could extend this further by more tightly integrating synchronous and asynchronous communication channels. For example, events within a room (e.g., significant changes to a room, or applet-specific events such as calendar scheduling) could generate electronic mail messages to inform others of the change. Similarly, workflow and commitment tracking can be tied to rooms containing the actual work being managed. What is more interesting is to see how the room metaphor can extend the capabilities of current asynchronous systems. For example, an “asynchronous” bulletin board applet could be used synchronously by a group to examine how a particular thread of conversation had evolved over time. Similarly, a project management applet could be created that tracks project tasks and dealines. Because it is in a room, people could use the same applet to track changes (asynchronously) or to discuss the project’s status in a meeting (synchronous).

5.3. Scenario: Flowing through the pre-, during-, and post-meeting process

This section illustrates through a scenario how a group using TeamWave can blend asynchronous and synchronous activities over the different phases of a meeting life-cycle.

Formal meetings are organized ahead of time. One example is a committee meeting, consisting of up to 10 people, and usually arranged for a formal exchange of information, for making decisions, or for delegating responsibilities (Jay 1976). Dubs and Hayne (1992) and Jay (1976) see this style of meeting as a process that cycles through three generic phases: pre-meeting setup, during-meeting activities, and post-meeting tear-down (which could lead into the next meeting). Each phase in turn contains a variety of sub-activities, as detailed by Dubs and Hayne (1992) and illustrated in Table 3. In unrelated work, Poltrock and Engelbeck (1997) have describes several meeting scenarios that closely match these phases.

Most groupware systems typically support only a single phase or activity within the meeting process. Scheduling software, for example, only schedules people and resources as part of the pre-meeting process. Similarly, email supports “informing” participants in the pre-meeting, and “communicating next steps” in the post meeting. Most real-time groupware provides single tools that support only particular tasks and processes in the during-meeting phase. Existing groupware contains considerable gaps that inhibit people from moving through the synchronous and asynchronous phases of the meeting process.

The room metaphor, in conjunction with email, can ease a group’s transitions across many (but not all) of these gaps. Consider a scenario inspired by a real life situation. Saul and Judy are co-chairs of the technical track of a conference, and Adam and Jeremy are the conference chairs. Saul is charged with setting up the first meeting for this team. Figure 4 illustrates how Saul can configure several rooms as part of the pre-meeting process. Saul begins by creating a room called “Meeting Room”. Using the roster applet, he then jots down the meeting goal and the roster of potential attendees. With the note organizer, he notes agenda points. He indicates the time of this and subsequent meetings on the calendar applet. Saul then collects information relevant to the agenda items. For agenda item 1, he creates another room called “Timeline”, and adds relevant information to it. A doorway to this room is included in the main meeting room, with an arrow attaching it to the agenda item. For agenda item 3, he uploads an external document (the ACM

Pre-meeting:	Setting goals	• review previous meetings to understand the status of the on-going process • describe meeting goals that establish a purpose to the meeting
	Getting participants	• develop roster of appropriate potential attendees • inform participants of the meeting
	Collecting materials	• develop and gather necessary documents, including the agenda • circulate background material ahead of time • select and book equipment, such as presentation tools or process aids
Meeting:	Start-up	• revise meeting objectives • review and revise agenda • revise tool selection
	During	• monitor activities to see how they conformance to meeting constraints e.g., time • follow particular processes and perform meeting tasks as required • record events as required
	Wind-up	• summarize the meeting • determine the next step
Post-meeting	Document	• create and distribute documents
	Communicate	• communicate next steps to participants

Table 3. Phases and activities in formal meetings. (Compiled from Dubs and Hayne, 1992)

Budget) so that others can retrieve and read it ahead of time. For agenda item 4, he includes a URL pointer to the old call for papers produced for the previous conference. (Alternatively, he could have used the groupware web browser to include the call for papers directly in the meeting space). He then uses the database applet to create an "action item" tool, where action items as well as who is responsible for carrying out the item can be added quickly during the meeting. Also included is a voting tool, which in this case is primed with a question about a potential agenda item, but which will also be used throughout the actual meeting. He then informs participants by email to tell them to look into this new room, and leaves a PostIt note in the room telling them what to do before the meeting. (Instead of email, the roster applet could conceivably be extended to automatically inform people about the meeting).

At this point, other participants can enter the room, review the information already in place, and add their own contributions (agenda items, tools, documents, etc.) in preparation for the meeting. In fact, the room in Figure 4 shows the roster tool partially filled in by participants who have indicated asynchronously that they will attend.

When the meeting actually begins, participants see one another "walk through the door" as their icons become visible on the room users list and as their images and telepointers appear within the room. Participants then work together synchronously. They review the agenda and move onto particular agenda items, using information that has been brought into the room ahead of time. For example, they move into the "Timeline" room when it is time to work on the conference schedule. Of course, new tools and information can be brought in as needed to support particular processes and tasks. Salient meeting points can be easily recorded, perhaps by inserting notes into the room, and by adding action items to the previously prepared database tool. Participants can then review these points and action items and summarize the meeting during the wind-up phase.

The post-meeting process, which is mostly asynchronous, is straight-forward as well. The state of the room becomes part of the meeting record, and the versioning system attached to rooms and applets allows people to review the evolution of meeting artifacts. Any participant can go into the room and retrospectively add any documentation and information that further summarizes the meeting and that leads into the next one. The room becomes the medium for the group to communicate asynchronously to each other.

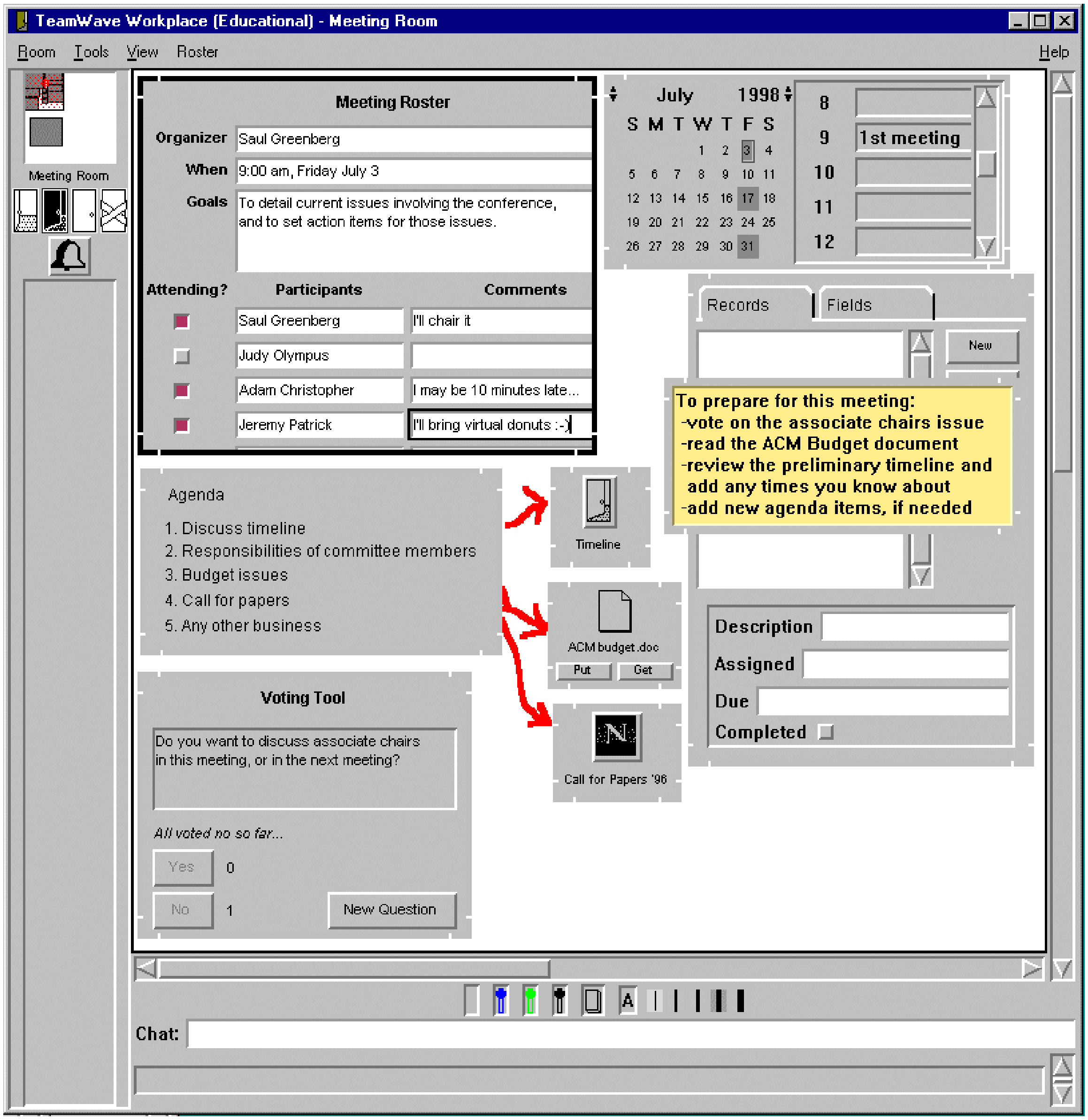


Figure 4. A TeamWave room configured for a formal meeting. As part of the pre-meeting process, the chairperson has indicated the meeting goals, the roster of participants, and has brought in materials and tools necessary for the meeting.

The room metaphor portrayed by TeamWave can only go so far to support formal meeting processes. Unlike many group decision support systems (Nunamaker, Dennis, Valacich, Vogel, and George 1991), a room does not impose any meeting process. Nothing dictates that rosters and agenda items be prepared, that relevant information is brought into the room ahead of time, or that action items are recorded. As with a physical room, it is the discipline and social structure of the team as well as the affordances of the space that creates a sense of a formal meeting room. Of course, formal protocols *could* be encoded into a room, where the software would insist that participants follow a particular meeting protocol.

In summary, this scenario shows how people using TeamWave rooms can move smoothly between synchronous and asynchronous activities within the meeting life-cycle. Because TeamWave artifacts persist, a room supports asynchronous activity. Participants can configure a meeting room ahead of time with necessary information and tools, leave messages for others, and add information retrospectively.

Because all items in a room have real-time groupware capabilities, TeamWave supports real-time conferencing just by having more than one person in a room. Participants can meet in a room and use it to review information, to bring up tools to support particular tasks, and to record items for posterity. Because TeamWave rooms are persistent and versioned, the room becomes a repository for all meeting artifacts and tracks how they evolved over time.

6. The Transition from Being Aware of Who's Around to Working Together in Real Time

In the real world, people regularly and serendipitously encounter one another and use these encounters as opportunities for further conversation. This is difficult to do in most groupware; there is a gap between knowing who is available in the electronic community, and using that knowledge to establish a real-time groupware session.

Successful teams rely on regular contact between their members, and this contact is often informal and unplanned (Kraut, Egido, and Galegher 1988; Root 1988; Cockburn and Greenberg 1993; Poltrock and Engelbeck 1997). In physical environments, informal interaction occurs regularly: people bump into each other in hallways; they see each other in the coffee room; they happen to be waiting together by the printer for their printouts; they reside in a live-in project room. While conversations may only last for a few seconds or minutes, much can occur within them. People coordinate actions (e.g., "I need to see you. When are you free?); they exchange information (such as requests for help: Poltrock and Engelbeck 1997); or they offer opportunities (e.g., "Come to our demonstration"). People can also see what others are doing, which lets them monitor progress and learn how things are done (Covi, Olson and Rocco 1998; Poltrock and Engelbeck 1997).

Yet the bottleneck to rich spontaneous interactions is distance (Kraut, Egido, and Galegher 1988), and users of wide area networks will be at a disadvantage unless a prosthesis that overcomes distance barriers is available. Many mundane factors interfere with making contact over computers. People must know electronic addresses and even machine names. People must ready software, equipment, and each other well in advance for real-time remote conferencing. With video conferencing and media spaces, people must be in the (usually few) conference rooms that have the media equipment available. But for informal interaction, people must find each other with minimal effort.

People must also select one or more of the many communication channels and applications that may be available to the group. From a technical perspective, sites may not have the same software; workstations may not support the necessary media (e.g., digital audio); specialized equipment may not be available (e.g., video cameras); poor networks may limit interactions; applications must run across platforms; and so on. From a human perspective, the communication channel or groupware must match a group's task (e.g. real time text editing), and accommodate how people are available (e.g. asynchronous vs. real-time). If people cannot make contact, then groupware systems—no matter how elegant—cannot be used.

6.1. How rooms support light-weight real time encounters

The room metaphor mitigates the transition of entering into a groupware conference. Because rooms are an inhabited space, they afford presence and awareness, encounters, habitation, and real-time meetings (Table 2). TeamWave also includes features that minimize technical hurdles, such as cross-platform compatibility.

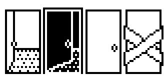
Being available. As mentioned previously, people can pursue their single user activities in TeamWave. As in a physical room used for both individual and group activities, people will be around more often and thus available for real time encounters (Covi, Olson and Rocco 1998). This is quite different from most groupware, where the only reason to be in a groupware session is for an explicit meeting.

Knowing who is around and available for interaction. When people populate a spatial setting, they sense who else is around in their local community as they walk through the corridors, glance into offices, and see others in public spaces. People judge other's availability for conversation by seeing if their door is open, by seeing how busy they look, and by a variety of other cues. A room metaphor can provide a similar sense of presence and awareness by displaying who else inhabits the rooms that comprise the electronic community, as well as status information about each person.

Identifying who else is around in the community encourages informal interaction. TeamWave does this in several ways: a user list, idle indicators and periodic video snapshots. *User lists* are shown in the "logged in users" window (Figure 2b), and include only the people who are currently logged on to a community's

server and thus reachable through TeamWave. Following the room metaphor, this is equivalent to wandering down the halls and seeing who is in their offices. Of course, being logged onto TeamWave does not guarantee that people are actually present, as they may be away from their computers. To compensate, an *idle indicator* located underneath each person's image indicates how long it has been since that person has used their keyboard or mouse. With short idle times, the person is almost certainly reachable. Longer idle times only estimates presence: that person may be away, or just not actively working on their computer (Greenberg 1996). In essence, the idle indicator tries to approximate the difference between someone being out of their office (and unavailable) even though their door is open, and someone being in their office. Next, in one experimental version of TeamWave *periodic video snapshots* were used to provide people with a better feel for the availability of others. The still images were replaced with snapshots taken several times per minute. This provides useful information about whether other people are actually present and available for collaboration, while still using very modest bandwidth. These snapshots bring a person's real room and environment into the virtual rooms, and serve a function similar to those found in the Portholes system (Dourish and Bly 1992).

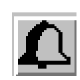
Other cues help decide availability. The presence of a person in a particular physical room, and the degree to which the door is open can indicate how interruptible that person may be. In TeamWave, this is mimicked by allowing people to see what room a person is in via the "Logged in Users" window (Figure 1b). More explicitly, one of the four *door states* of a room, which can be set by a room's occupants, is



visible on the "Rooms on this Server" window (Figure 1a). As in real life, a wide and partially open door icon indicates a willingness to accept interruptions, while a barred door suggests that the room (and the people in it) is inaccessible[5].

Finally, special rooms can be set aside as informal meeting places. For example, one site had a lounge that used the image applet to point to popular daily comic strips available on the Web, such as the "Dilbert of the Day." This became a popular room, which afforded accidental encounters.

Establishing contact. Once a person decides that someone is available, there are several ways of actually establishing contact with them. Congruent with the room metaphor, one could just barge into a room. This is, of course, subject to the group's norms and social mores as it relates to the room and the people within them. Similar to real life, the other room occupants sees that a person has entered the room because their picture appears (Figure 1h). They can start talking to one another through the chat facility available in every room (Figure 1o). Though we would hardly expect a lengthy meeting to be carried out in a room without the benefit of an audio channel, text can be useful for short or sporadic interactions, or where it is impractical to provide an audio connection. If a person would rather initiate a conversation before entering a room, they can *page* anyone on the user's list (Figure 2b). This displays a note on the other person's screen (Figure 2c), which can be quickly answered to (say) invite that person in. Any person's phone number can be found in their business card (Figure 1c), allowing phone calls to be quickly established made before a room is entered, or afterwards. Once in a room, a person can attract the attention of others in the room (some who may not be looking at their computer screen) by ringing the *bell* (Figure 1g), which plays a sound on all computers.



Of course, one person may want to establish contact with others who are not currently logged on to TeamWave. In this case, the person can revert to the asynchronous tools mentioned in previous sections, such as leaving a note in a room to suggest a meeting time and place.

Working together. The power of the room metaphor is that, once in a room, the working context is immediately available. All tools and room artifacts are ready to hand, and new tools are easily added. In TeamWave the back wall is a whiteboard, so people can augment their conversation at any time simply by sketching, where they can draw new artifacts and annotate existing ones (Greenberg, Hayne and Rada 1995). Thus "back of the envelope" style conversations are easy. Similarly, the ever-present chat tool means that conversation (albeit low bandwidth) is always possible. As well, the applets within a room can quickly become the focus of conversation.

[5] Door states also serve to mediate access to rooms that have no one in them: open doors indicate more public rooms, whereas closed doors indicate more private rooms.

Rooms also replace the rigid concept of a "meeting" enforced by most groupware systems. Many such systems have session managers, where people create real-time meetings and attend them (Greenberg and Roseman, in press). In contrast, a meeting in TeamWave is a social phenomenon created by the group simply by being in the same room at the same time (Table 2).

6.2. Example: A serendipitous encounter

A simple but real example serves to illustrate how all these features work together. We had hired a student to work in our laboratory part-time for a month. Because of summer vacations, scheduling differences, and the part time nature of the student's work, the laboratory supervisor (Greenberg) had not actually had a chance to talk to the student. While Greenberg was tele-working in his TeamWave room from home one day (which is 110 kilometres from the laboratory), he noticed through the awareness facilities that the student was also working in another room. Because she was in a room actively used by the team, he felt free to enter it. They then had their first conversation. It began with social banter, but progressed to the student discussing and presenting the work she was doing. This included a tour of her working contributions to the room and its work artifacts. After the conversation, they both went back to their individual work in their separate rooms.

In summary, TeamWave's room metaphor affords opportunistic and light-weight interaction, as illustrated by the example and summarized in Table 2. The social opportunity to meet can be by circumstance and serendipity rather than by intention and planning. We saw in the example that both people were around (habitation) because they were pursuing their individual work; they were aware of one another through the awareness tools (presence and awareness); one person easily initiated the conversation just by entering the room (encounters); and both found it easy to incorporate the room context and artifacts into the discussion (meeting definition). Everything was extraordinarily light-weight: the only "required" action to initiate the transition to a real time interaction was a single mouse click by Greenberg to enter the room.

7. The Transition From a Technical Space to a Social Place for Work

So far we have seen how the room metaphor lets us apply characteristics of physical spaces to virtual spaces. Yet thinking about the characteristics of space alone does not capture all the richness of these environments . In the physical world, a space is often adopted by a group, and transformed into a *social place* for their interactions. For example, a house becomes a home when a family lives in it; a generic office becomes a personal one when its occupant hangs pictures on the walls, clutters the desk and occupies it while they do their work; a room becomes a project room when a group uses it as a place to pursue their activities and to develop project artifacts (Covi, Olson and Rocco 1998). In this section, we will see how the room metaphor can serve as a basis for forming such social place.

7.1. The difference between spaces and places

Harrison and Dourish (1996) argue that designers who think only about the features of physical spaces are missing a crucial aspect of collaboration. Whereas *space* provides a physical venue giving opportunities for collaboration, it is the group's understanding of how the space should be used that turns it into a social *place*. Places extend spaces by including the social meanings of actions within a space, the cultural norms and mores that set convention, as well as the group's cultural understandings of the roles played by the people and artifacts that inhabit the space. Harrison and Dourish (1996) reason that a group forges a sense of place over time as its members actively participate in the space and appropriate objects within it. Places are thus social constructs, the "understood reality" of the group that derives (in part) from the opportunities afforded by the space. Benford, Brown, Reynard and Greenhalgh (1996, p81) also agree with this distinction, when they define space as "a context which provides a consistent, navigable and shared spatial frame of reference (e.g., Cartesian space)", and a place as a "basic containing context for participants".

Fitzpatrick, Mansfield, and Kaplan's (1996) *locales framework* uses Strauss' theory of social worlds (Strauss 1993) to distinguish social places from technical spaces. In this framework, a *social world* is a group of people with some common purpose, a site for collaboration, and some means to communicate. A *locale* is the actual site in which a group collaborates, the actual means by which people communicate, and the actual means by which the work is achieved[6]. For example, a team with the goal of designing a software system would form the social world. If the team met in a meeting room, the room and all its artifacts—its

[6] Definitions and example provided by Tim Mansfield and Geraldine Fitzpatrick, personal communication.

visual and auditory communication, its whiteboard, its table, and the paper within it—would form a *physical locale*. If instead the team met through a groupware system in conjunction with a telephone, then that would form a *virtual locale*. While a virtual locale may not be actually realized in physical space, it still provides the site and means for the social world to collaborate.

More formally, the locales framework comprises five aspects, as summarized here (Fitzpatrick, Mansfield, and Kaplan 1996).

- *Locale foundations* define a collection of people and artifacts (tools, objects, information) in relation to the central purpose of the social world. A locale within a social world is best considered as a "center" of collective purpose that is part of a dynamic and continually evolving system. Locales are fluid places with social meaning that may be mapped onto physical spaces, although care must be taken that the structure of the physical space does not conflict with the dynamic social structures.
- *Mutualit*y considers those interactions within locales that maintain a sense of shared place. Mutuality includes "presence" information that people and artifacts make available to others, and how people maintain awareness of that information. It also includes "capabilities" that entities have to transmit and receive information, and how entities choose from these capabilities to create a particular presence-awareness level.
- *Individual view over multiple locales* acknowledges that individuals can be participating in many locales. Each person's individual view is an aggregation of their views onto their particular locales. People also manifest a *view intensity* onto particular locales as he or she varies their focus and participation across locales.
- *Interaction trajectories* concern how courses of action evolve over time. In essence, people come into locales and social worlds with past experiences, plans, and actions. The trajectory describes how these move through time, for example, as people negotiate plans and actions with each other.
- *Civic structures* concern how interactions fit within a broader communal level. Civic structures can be considered a "meta-locale" that describe how social worlds and locales relate to one another, how people find their way between them, and how new locales are formed and old ones dissipated.

All aspects of the locales framework describe characteristics of what makes a social place. From these characteristics, we can then consider how these aspects are afforded by technical spaces such as TeamWave's room metaphor.

7.2. Constructing a social place within TeamWave Workplace rooms

Fitzpatrick, Mansfield, and Kaplan (1996) have argued that groups can create a social place that is not bounded to a physical space. While this is true, we suggest that the room metaphor and its spatial properties can assist the distributed group in fashioning a social place within an electronic medium. In essence, a newly formed TeamWave Workplace site is a space that lacks social context. However, the affordances of TeamWave rooms (Table 2) and the way rooms can be customized means that the individuals comprising a social world can conveniently craft the rooms into a locale that acts as their social place.

Community membership and formation. The TeamWave administrator controls community and group membership. The act of deciding who is allowed to enter a particular TeamWave site defines (albeit in a technical manner) who belongs to that community. This may comprise predefined groups (e.g., all members of a department), distributed special interest groups (e.g., people with interests in a particular topic), goal-oriented teams (e.g., members of a conference committee), or some ad-hoc membership (e.g., individuals interested in visiting a particular site).

Of course, a community is more than an access control list. In practice, formation of the initial community may require some champion or sub-group to motivate other members to actually enter and participate within the TeamWave site. This could include handholding to make sure that the software is correctly installed and configured, some training, and probably a strong motivating factor. For example, we saw a community form itself when one member of a TeamWave mailing list offered to set up a TeamWave site for other list members. He motivated others by offering hands-on experience for early TeamWave adopters, as well as visits by TeamWave experts who could provide the group with guidance. In his own words:

> …I don't know if there is anyone else in this list who is currently in the learning/investigating stage of working with collaborative software but I thought I'd offer this idea. I recommend that [we] establish a periodic, ongoing "conference" hosted by an experienced TeamWave developer/facilitator where

potential users could log in and participate in a way that would exercise all of the capabilities of TeamWave…

After receiving positive responses, the TeamWave site was set up and other list members were invited to attend. Some rooms were created ahead of time (preliminary locales), and notes were sent around that included explicit mores on how they should be used (which could be considered the initial formation of a civic structure). Part of this structure included an explicit way to set up meeting times i.e., a way to increase mutuality.

When you first login, you will find a reception area or lounge, with doors to other offices. Each of you will have an office for your own investigations. There will be a door to a meeting room too. I will make the lounge the default and leave a group-scheduling calendar and some notes. Please use the calendar to indicate your dates and times on-line to assist others in working with you. May I suggest using the meeting room for meetings to keep the lounge uncluttered and faster loading. The meeting room could be used for your feedback to Mark and his group.

What is particularly interesting about this example is that much of the correspondence involved the social construction of a place, who would belong to it, and the initial social rules and conventions. Indeed, the community as well as its social place was in some sense established well before anyone actually entered into the TeamWave space. Even so, the room metaphor began to come into play. It gave the group a space to meet that they could call their own. Because rooms could be customized, the initial structure was tailored to fit group needs. As well, rooms provided a context for the expected behaviors and the mores of the group.

Using rooms to create locales. Rooms are just bounded spaces. Its physical entities are its walls and tools, the applets, and the information contained in the applets. Its personal entities are the people that come and go in a room. A person or group can use a room as a bounded center of a social world. We have already seen examples of this in previous sections. The personal room of Figure 3 is a locale for an individual, although its membership can be expanded as people are invited into that person's place. The meeting room of Figure 4 becomes the locale for a specific group's ongoing meeting activities. One could argue that the set of rooms used by that group is a locale, where individual views shift over them as the focus of the group changes. The point is that it is the room's users that craft the meaning of a room as a social place. Much of this is done by manipulating the room artifacts, and by customizing its contents to fit the social needs of the group.

Other aspects of rooms afford other criteria included in the locales framework. Mutuality is afforded by the various awareness widgets, by telepointers, and by the immediate feedthrough of changes when more than one person are in a room. Interaction trajectories are afforded by the persistence of artifacts within a room, as well as the ability to revisit previous versions. Civic structures are afforded by the ability to leave notes within a room that spell out the terms of engagement and that directly reference objects within the room by proximity or explicit pointers.

Herlea (1997) gives an example of how a multi-room locale can be created. Her goal was to create a locale for iterative requirements engineering based on the soft systems requirements methodology, where the group's collective purpose would be loosely structured by the locale. Figure 5 illustrates this. In this top-level view, we see a room that contains sets of doorways to other rooms. These sets of rooms are configured to support various aspects of the methodology. At the top left, we see three rooms stepping through phases of requirements discovery, which includes rooms for scope definition, brainstorming, and end-user requirements. The actual rooms are pre-configured with tools and documentation appropriate for the work. Other room sets include requirements refinement (top right), viewpoints analysis (bottom right), and solution selection (bottom left). At the center are "social" rooms, including a meeting room, coffee room, and agenda room. The pre-constructed locale becomes a guiding structure that supports their social world. That is, the locale helps the group pursue its collective purpose of requirements engineering. As individuals navigate this space, they are changing their view intensity.

The caveat to TeamWave rooms as locales is that rooms are at best only approximations of locales. Because the boundaries of a room are fairly rigid, they act as a container rather than a center. People cannot be in more than one room at a time, even though they may have an individual view that perceives a locale as a union of several rooms. To capture the full richness of the locales framework, rooms would have to be more fluid and permeable than they are. While people can copy applets and their contents from room to

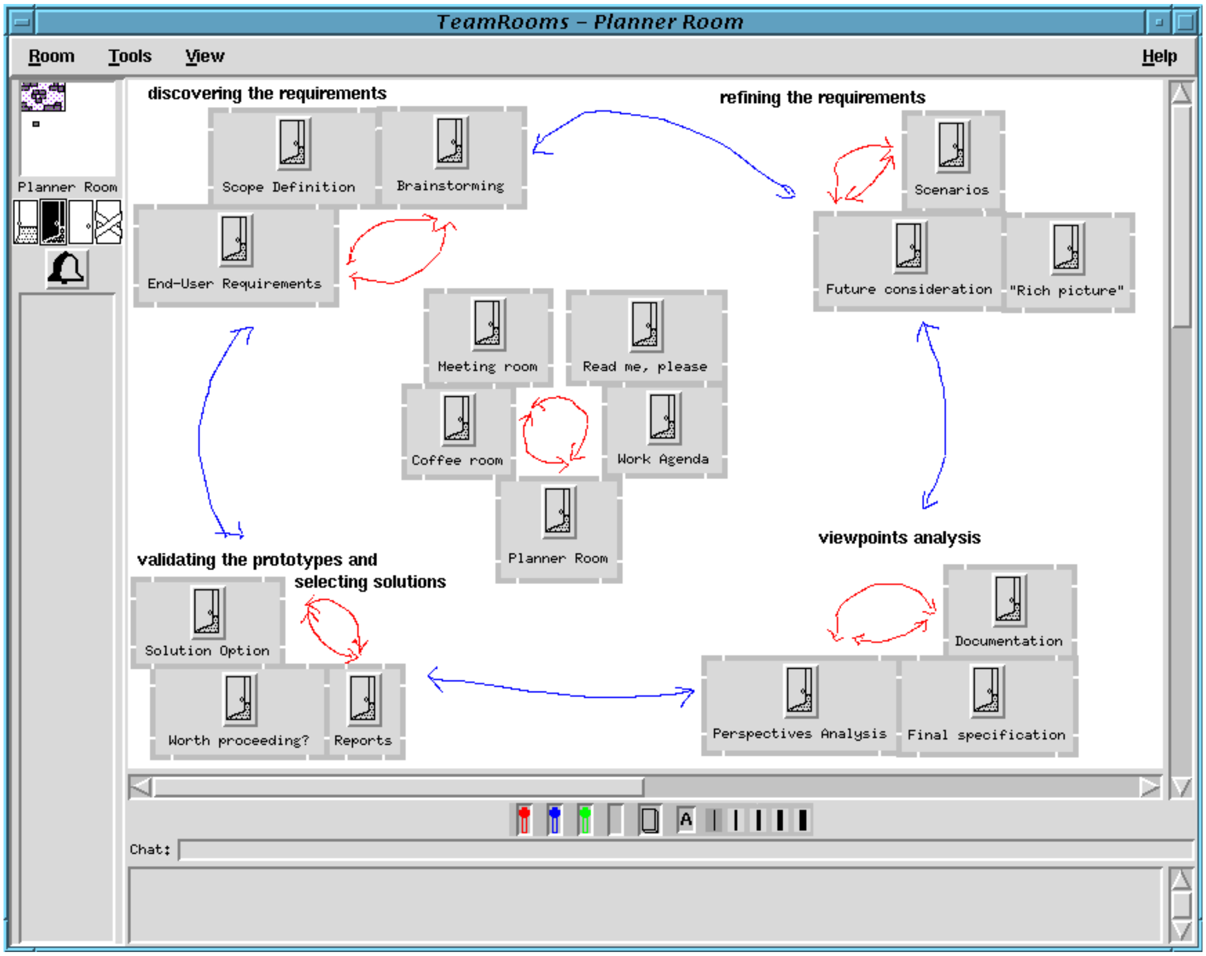


Figure 5. The LoReN environment. Rooms are configured to support the iterative requirements engineering cycle. From Herlea (1997), with permission.

room, the physical space is fairly static and somewhat heavyweight compared to the dynamics of a social place.

In summary, the room metaphor can help a group form a social place. Rooms serve as a destination where people can agree to meet and work together. Because rooms can be customized and because they persist, they can be crafted into a locale that evolves over time. Proximity and action implies that people are aware of what others are doing in a room, which affords mutuality. Because rooms are often related to one another, locales can span across a series of rooms. Because rooms are revisited over time, and because "rules of engagement" can be posted in a room, civic structures will develop over time. All these ease the transition from electronic spaces to social places.

8. Related Work

The value of spatial metaphors are well known, and have long been used as a foundation for interface design (Kuhn and Blumental, 1996). In 1986, for example, an elegant room-based system called "Rooms" was built to support task-switching, where each room acted as a container for a particular set of tasks (Henderson Jr. and Card 1986). There are a now variety of collaborative systems that rely on some sort of spatial metaphor. We briefly describe a few of the approaches and systems here, loosely categorizing them as meeting points, social spaces, and working spaces.

Meeting points. Several systems use the concept of "place" or "room" as a gathering point for collaborators. The idea is that when several people congregate at the same virtual point, other tools are automatically invoked to connect them together.

With media spaces (e.g., Abel 1990), physical rooms are connected via always-on video and audio channels. From one physical room, a person can look into another physical room, greet the people located there, and start conversations. This combines their physical spaces together, albeit in a limited manner.

Virtual rooms as meeting points are also popular. The CAVECAT media space (Mantei, Baecker et al 1991) used the idea of walking into the same virtual room (as displayed on a screen) as another user as a mechanism to initiate a audio/video connection. The DIVA Virtual Office Environment (Sohlenkamp and Chwelos 1994) and GroupKit's Rooms session manager (Roseman and Greenberg 1996c) used virtual rooms to gather and organize people and their documents. Users in these rooms launch external groupware editing tools. However, these tools are separate from the room itself and run in their own windows.

However, these types of systems do not act as an integrated bounded space. Rather, they mostly afford contact facilitation, informal conversation, and application launching. In contrast, TeamWave integrates the communication channels, the tools, and the way persistent artifacts are created as part of the meeting place by using rooms as containers for all these activities.

Social spaces. Other systems create social spaces for conversational interaction. The simplest of these are chat rooms, where users connect to a room and can chat to others in the room with typed messages (Turkle 1995). Recent chat rooms found on online services have added rudimentary graphics, but the interaction is still entirely text-based. Another popular class of social space is Multi-User Dungeons (MUDs and MOOs). Born of text-based adventure games, traditional MUDs are text-based systems where users connect to a central server. The server hosts a variety of different rooms, each with a unique description and set of objects. As with chat rooms, a person can enter any number of different rooms and chat with other people in those rooms. However, they can also type commands to create and modify objects in the rooms.

Several systems have augmented MUDs with non-textual tools. For example, the Jupiter project (Curtis and Nichols 1993) added MBONE audio and video conferencing and graphics capabilities through shared whiteboards. Multi-user virtual environments (MUVEs) create a graphical world, either in 2 ½ or 3 dimensions, where people can navigate the world and encounter others (e.g., DIVE, Carlsson and Hagsand 1993). People are often represented as avatars i.e., graphical automations (which may even include a video face) that can be seen by others. Text or audio connections to others are often triggered by proximity. If one person's avatar is close to another, they can hear each other.

While MUDs and MUVEs have been used in limited ways to support collaborative work, most of their appeal is for social interaction. They tend to support large, loosely-knit virtual communities, where anyone can enter the space, rather than smaller, goal-oriented working groups. Essentially, this is the difference between software supporting virtual communities and 'teamware' products (including TeamWave) that are oriented towards modest-sized teams and organizations (Wong, 1998).

Working spaces. There are a variety of systems developed around a rooms model that contains artifacts directly supporting work activities of small groups.

- *CoLab* was a face to face meeting support system that gave a group several meeting tools (Stefik, Foster, Bobrow, Kahn, Lanning, and Suchman 1987a). After seeing some problems in the software, the designers discussed what it would be like to embed the tools within a room metaphor (Stefik, Bobrow, Foster, Lanning and Tatar 1987b). While the metaphor was not implemented, this was one of the first efforts that seriously considered the design and implications of a room metaphor to groupware.
- *Mushroom Project* is a software framework that provides a type of room called *Mrooms* (Kindberg 1996). Mrooms contain representations (images) of the users who are in a room, communication tools, and information about objects that users share. Objects, which include whiteboards, documents, and multimedia presentations are usually displayed in separate windows.
- *Collaborative Virtual Workspace (CVW)* by Mitre Corporation divides a building into floors and rooms (Spellman, Mosier, Deus and Carlson 1997). Rather than building applets from scratch, CVW integrates external tools and services into a single environment, and these are mostly displayed in separate windows. However, CVW developers have created their own tools when necessary.
- *Kansas* is an object-oriented multi-user graphical environment written in the Self language (Smith, Wolczko, and Ungar 1997). Designed as a virtual space for learning, Kansas is a large, flat plane rather than a room. Users can be located anywhere within it, and they will see a local portion of

Kansas (but radar overviews lets people see where others are). Users can also program and run graphical objects within it. When a user alters the properties of Kansas, all other users feel the effects. Similar to some MUVEs, moving together and apart in Kansas will make and break audio connections among users.

- The three versions of *Orbit* (Mansfield, Kaplan, Fitzpatrick et. al. 1997) and its predecessor *wOrlds* (Tolone, Kaplan and Fitzpatrick (1995) are implemented prototypes that test how the locales theory can be supported by software. These systems try to provide a highly flexible way of viewing and shifting between multiple locales and the shared objects contained by them. This innovative approach differs considerably from the room metaphor, for it eschews the relatively fixed way that rooms contain objects, and does not preserve the spatial relations between objects.
- *CommonPoint* was an unreleased collaborative desktop environment from Taligent (described in Orfali, Harkey and Edwards 1996). The desktop uses a "People, Places, and Things" metaphor to provide very rich network places where people can gather and work with shared documents. However, CommonPoint would have required radical changes to the computing environment, completely replacing conventional operating systems and their desktop metaphors.
- A variety of commercial groupware products now use the notion of a room as a bounded space: Instinctive Technology's *eRoom 1.0*, *facilitate.com 4.5*, Changepoint's *involv Intranet 2.0*, and Lotus' *InstantTeamRoom 1.0.* They are geared primarily toward asynchronous communication, and provide almost no support for real-time collaboration, even when users are working the artifacts in the same room (Wong 1998).

In summary, while these systems rely on spatial metaphors to some degree, they vary greatly in how they support the features mentioned in Table 2. For example, because the Mushroom Project and CVW display tools in separate windows on the screen, users cannot relate them by spatial proximity as they can in TeamWave. Orbit and wOrlds stress highly tailorable views against the idea of a room as container. Kansas uses an unbounded space rather than bounded rooms. Some systems support mainly asynchronous collaboration.

9. Summary

In this paper, we described how a room metaphor can allow groupware to ease transitions across different styles of work. We listed a variety of features afforded by the room metaphor, and how a group can leverage these for individual and collaborative work (Table 2). We described TeamWave Workplace and its particular implementation of a room metaphor. Four groupware gaps were then presented, and we showed how transitions across these gaps can be reduced by the properties of the room metaphor. First, the distinction between single user and groupware applications is removed by making rooms suitable for both individuals and groups, and by making the shift between an individual or group space just a function of the number of people in a room. Second, people can move fluidly between synchronous and asynchronous work when room artifacts are persistent, and when people work together imply by occupying the same room at the same time. As an example, we showed how people can move through the pre, during, and post meeting processes of a formal meeting by allowing rooms to be configured with appropriate tools and information ahead of time, by having all tools support real time interaction, and again by making artifacts in a rooms persist over these meeting phases. Third, the difficulty of establishing real-time contact is reduced by providing people with awareness of who is around for casual interaction, and by letting them meet by entering a common room. Fourth, we showed how a technical space can become a social place by the way a community is formed across rooms, and by the way people within a room craft meaning into it.

Our understanding of the room metaphor for group work is impoverished, as research is still in its early stages. For example, we need to evaluate how teams use virtual rooms. Yet this is difficult to do (Grudin 1989). Laboratory and short term experiments will not reflect the way a team uses and shapes its rooms over time. Neither are we ready to perform field studies: commercial-grade groupware systems using room metaphors are rare and the few existing ones have been deployed only recently. We also have to recognize that even small design and implementation deficiencies in current systems can greatly affect how teams adopt and use them.

We have a ways to go before we realize the full potential of the rooms metaphor. Still, the benefits of rooms are very promising, as shown in this paper and in the TeamWave case study.

10. Software Availability

Information on TeamWave Workplace, including directions on how to license the software, is available at http://www.teamwave.com. Information on related research and software (such as the GroupKit groupware toolkit) is available through University of Calgary's GroupLab Web site at http://www.cpsc.ucalgary.ca/grouplab/.

11. Acknowledgements

Many people on the GroupLab team contributed in one way or another to this work. This includes Carl Gutwin and his work on awareness, and Ted O'Grady and his work on flexible toolkit infrastructures. Alex Mitchell (along with Mark Roseman) helped code a very early rooms-based session manager. Intel Corporation and the National Science and Engineering Research Council of Canada funded this research. We are particularly grateful to the support and encouragement provided by Intel's Jim Larson, Kenrick Mock, and Tony Salvador.

12. References

Abbot K. and Sarin S. (1994). Experiences with workflow management: Issues for the next generation. In *Proceedings of the ACM CSCW'94 Conference on Computer Supported Cooperative Work*, pp. 113-120, ACM Press.

Abel M. (1990). Experiences in an exploratory distributed organization. In J. Galegher, R. E. Kraut, and C. Egido (Eds.), *Intellectual Teamwork: Social Foundations of Cooperative Work*, pp. 489-510. Lawrence Erlbaum Associates, Hillsdale, New Jersey.

Alexander, C. (1979). *The timeless way of building.* Oxford University Press.

Baecker R. (1993). *Readings in groupware and computer-supported cooperative work: Assisting human-human collaboration.* Morgan-Kaufmann, San Mateo, CA.

Baecker R. M., Nastos D., Posner I. R., and Mawby K. L. (1993). The user-centred iterative design of collaborative writing software. In *ACM INTERCHI Conference on Human Factors in Computing Systems*, pp. 399-405, Amsterdam, April 24-29, ACM Press.

Baecker R., Grudin, J., Buxton, W., and Greenberg, S. (1995). *Readings in Human Computer Interaction: Toward the Year 2000*. Morgan-Kaufmann, San Mateo, CA.

Benford S., Brown C., Reynard G., and Greenhalgh C. (1996). Shared spaces: Transportation, artificiality, and spaciality. In *Proceedings of the ACM CSCW'96 Conference on Computer Supported Cooperative Work*, Boston, USA, ACM Press.

Buxton W. A. S. (1992). Telepresence: Integrating shared task and person spaces. In *Proceedings of Graphics Interface '92*, pp. 123-129, Morgan Kaufmann. Reprinted in Baecker (1993).

Card S. and Henderson Jr D. (1987). A multiple virtual-workspace interface to support user task switching. In *Proceedings of the ACM SIGCHI+GI 1987 Human Factors in Computing Systems and Graphics Interface*, pp. 53-59, Toronto, ACM Press.

Card S. K., Robertson G. G., and Mackinlay J. D. (1991). The information visualizer, an information workspace. In *Proceedings of ACM CHI'91 Conference on Human Factors in Computing Systems*, pp. 181-188.

Carlsson, C. and Hagsand, O. (1993). DIVE - A platform for multi-user virtual environments. *Computers and Graphics*, 17(6), pp. 663-669.

Cockburn A. and Greenberg S. (1993). Making contact: Getting the group communicating with groupware. In *Proceedings of the ACM COOCS Conference on Organizational Computing Systems*, Milpitas, California, November 1-4, ACM Press.

Cockburn A. and Thimbleby H. (1991). A reflexive perspective of CSCW. *ACM SIGCHI Bulletin*, **23**(3), pp. 63-68, July.

Conklin J. (1988). gIBIS: A hypertext tool for exploratory policy discussion. In *Proceedings of the Conference on Computer-Supported Cooperative Work (CSCW '88)*, pp. 140-152, Portland, Oregon, September 26-28, ACM Press.

Conklin E. J. (1992). Capturing organizational memory. In D. Coleman (Ed.), *Groupware '92*, pp. 133-137. Morgan Kaufmann. Reprinted in Baecker (1993).

Covi, L., Olson, J. and Rocco, E. (1998). A room of your own: What do we learn about support of teamwork from assessing teams in dedicated project rooms? *Proceedings of CoBuild'98 First International Workshop on Cooperative Buildings: Integrating Information, Organization, and Architecture*, February 25-26, GMD, Darmstadt, Germany.

Curtis P. and Nichols D. (1993). MUDs grow up: Social virtual reality in the real world. In *Proceedings of the Third International Conference on Cyberspace*.

DeSanctis, G., and Gallupe, B. (1987). A foundation for the study of Group Decision Support Systems. *Management Science* 33(5), p589-609.

Dix, A., Finlay, J., Abowd, G. and Beale, R. (1993). *Human Computer Interaction.* Prentice Hall (UK), Hertfordshire, UK.

Dourish P. and Bly S. (1992). Portholes: Supporting awareness in a distributed work group. In *Proceedings of the CHI'92 Conference on Human Factors in Computing Systems*, pp. 541-547, Monterey, May 3-7, ACM Press.

Dubs S. and Hayne S. (1992). Distributed facilitation: A concept whose time has come? In , *Proceedings of ACM CSCW'92 Conference on Computer-Supported Cooperative Work*, pp. 314-321.

Fitzpatrick G., Mansfield T., and Kaplan S. (1996). Locales framework: Exploring foundations for collaboration support. In *Proceedings of the OzCHI'96 Sixth Australian Conference on Computer-Human Interaction*, pp. 34-41, Hamilton, New Zealand, November 24-27, IEEE Computer Society Press.

Flores F., Graves M., Hartfield B., and Winograd T. (1988). Computer systems and the design of organizational interaction. *ACM Transactions on Office Information Systems*, **6**(2), pp. 153-172. Republished in Baecker (1993).

Greenberg S. (1990). Sharing views and interactions with single-user applications. In Proceedings of the ACM/IEEE Conference on Office Information Systems, pp. 227-237, Cambridge, Massachusets, April 25-27.

Greenberg S. (1996). Peepholes: Low Cost Awareness of One's Community. In *Companion Proceedings of the ACM SIGCHI'96 Conference on Human Factors in Computing System*, pp. 206-207, Vancouver, Canada, April, ACM Press.

Greenberg S., Gutwin C., and Roseman M. (1996). Semantic telepointers for groupware. In *Proceedings of the OzCHI '96 Sixth Australian Conference on Computer-Human Interaction*, Hamilton, New Zealand, November 24-27.

Greenberg S., Hayne S., and Rada R., eds (1995). *Groupware for Real-Time Drawing: A Designer's Guide*, 248 pages, McGraw-Hill Book Company Europe, Berkshire, England.

Greenberg S. and Roseman M. (1996). GroupWeb: A WWW Browser as Real Time Groupware. In *Companion Proceedings of the ACM SIGCHI'96 Conference on Human Factors in Computing System*, pp. 271-272, Vancouver, Canada, April, ACM Press.

Greenberg, S. and Roseman, M. (in press). Groupware toolkits for synchronous work. in M. Beaudouin-Lafon, editor, *Computer-Supported Cooperative Work*, Trends in Software Series, John Wiley & Sons Ltd.

Grudin, J. (1994). Computer Supported Cooperative Work: History and focus. *IEEE Computer*, 27(5), p19-26.

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

Gutwin, C. and Greenberg, S. (1996). Workspace awareness in real-time distributed groupware: Framework, widgets, and evaluation. In R. J. Sasse, A. Cunningham, and R. Winder (Eds.), *People and Computers XI (Proceedings of the HCI'96)*, pp. 281-298. Springer-Verlag. Conference held at Imperial College, London, August 20-23.

Harrison S. and Dourish P. (1996). Re-place-ing space: The roles of place and space in collaborative systems. In *Proceedings of the ACM CSCW'96 Conference on Computer Supported Cooperative Work*, Boston, USA, ACM Press.

Herlea, D. (1997). *A groupware system for negotiating software requirements.* M.Sc. thesis, Dept of Computer Science, University of Calgary, Alberta, Canada.

Henderson Jr D. and Card S. (1986). Rooms: The use of multiple virtual workspaces to reduce space contention in a window-based graphical user interface. *ACM Transactions on Office Information Systems*, **2**(1), pp. 211-243.

Ishii H. (1990). TeamWorkStation: Towards a seamless shared space. In *Proceedings of the Conference on Computer Supported Cooperative Work (CSCW '90)*, pp. 13-26, Los Angeles, California, October 7-10, ACM Press.

Ishii H., Kobayashi M., and Grudin J. (1993). Integration of interpersonal space and shared workspace: Clearboard design and experiments. *ACM Transactions on Information Systems*. October. Reprinted in S. Greenberg, S. Hayne and R. Rada (1995), pp. 96-125.

Jay, A. (1976). How to run a meeting. *Harvard Business Review*, 54(2), March/April. Reprinted in Baecker (1993).

Johansen, R. (1988). *Groupware: Computer Support for Business Teams.* The Free Press.

Kindberg, T. (1996). Mushroom: a framework for collaboration and interaction across the Internet. *The International Workshop on CSCW and the Web*, Sankt Augustin, Germany, February. http://www.dcs.qmw.ac.uk/research/distrib/Mushroom/CSCWWeb.html

Kraut, R., Egido, C. and Galegher, J. (1988). Patterns of contact and communication in scientific collaboration. In *Proceedings of the Conference on Computer-Supported Cooperative Work (CSCW '88)*, pp. 1-12, Portland, Oregon, September 26-28, ACM Press.

Kremer, R. (1997). *A concept map meta-language.* PhD Thesis, Department of Computer Science, University of Calgary, Calgary, Alberta, Canada.

Kuhn, W. and Blumental, B. (1996). Spatialization: Spatial metaphors for user interfaces. ACM Conference on Human Factors in Computing Systems, Tutorial Notes. Also available as GeoInfo 8, Dept of Geoinformation, Technical University of Vienna, Austria.

Lange B. (1992). Electronic group calendaring: Experiences and expectations. In D. Coleman (Ed.), *Groupware '92*, pp. 428-432. Morgan Kaufmann. Reprinted in Baecker (1991).

Leiva-Lobos, E., De Michelis, G. and Covarrubias, E. (1997). Augmenting and multiplying spaces for creative design. In *Proceedings of the Group'97 International Conference on Supporting Group Work*, pp. 177-186, ACM Press.

Malone T. W., Lai K. Y., and Fry C. (1992). Experiments with Oval: A radically tailorable tool for cooperative work. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW'92)*, pp. 289-297, Toronto, Ontario, October 31 - November 4, ACM Press.

Mansfield, T., Kaplan, S., Fitzpatrick, G., Phelps, T., Fitzpatrick, M. and Taylor, R. (1997). Evolving Orbit: A progress report on building locales. In *Proceedings of the Group'97 International Conference on Supporting Group Work*, pp. 241-250, ACM Press.

Mantei M., Baecker R., Sellen A., Buxton W., Milligan T., and Wellman B. (1991). Experiences in the use of a media space. In *Proceedings of the ACM CHI'91 Conference on Human Factors in Computing Systems*, pp. 203-208, ACM Press.

Nardi, B. (1991). Twinkling lights and nested loops: Distributed problem solving and spreadsheet development. In Greenberg, S. (ed) *Computer Supported Cooperative Work and Groupware.* pp. 29-52, Academic Press, London.

Nunamaker, J., Dennis, A., Valacich, J., Vogel, D. and George, J. (1991). Electronic meeting systems to support group work. *Communications of the ACM* 34(7), 40-61.

O'Grady T. and Greenberg S. (1994). A groupware environment for complete meetings. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems: Posters and short papers*

Volume 2, pp. 307-308, Boston, ACM Press.

Orfali R., Harkey D., and Edwards J. (1996). *The Essential Distributed Objects Survival Guide*, John Wiley and Sons.

Poltrock, S. and Engelbeck. G. (1997). Requirements for a virtual collocation environment. In *Proceedings of the Group'97 International Conference on Supporting Group Work*, pp. 61-70, ACM Press.

Preece, J. (1994). *Human Computer Interaction.* Addison-Wesley.

Rein G. L. and Ellis C. A. (1991). rIBIS: A real-time group hypertext system. *International Journal of Man Machine Studies*, **34**(3), pp. 349-368, March. Republished in Greenberg 1991.

Root, W. R. (1988). Design of a multi-media vehicle for social browsing. In *Proceedings of the Conference on Computer-Supported Cooperative Work (CSCW '88)*, pp. 25-38, Portland, Oregon, September 26-28, ACM Press.

Roseman M. (1996). Managing complexity in TeamRooms, a Tcl-based Internet groupware application. In *Proceedings of the 1996 Tcl/Tk Workshop*, Usenix Press.

Roseman M. and Greenberg S. (1997). Simplifying component development in an integrated groupware environment. In *Proceedings of the ACM UIST'97 Symposium on User Interface Software and Technology*, Banff, Alberta, ACM Press.

Roseman M. and Greenberg S. (1996a). TeamRooms: Network places for collaboration. In *Proceedings of ACM CSCW'96 Conference on Computer Supported Cooperative Work*, Boston, USA, ACM Press.

Roseman M. and Greenberg S. (1996b). A tour of TeamRooms. *Technical Video Track of the ACM Conference on Human Factors in Computing Systems*. Videotape, 8.4 minute.

Roseman, M. and Greenberg, S. (1996c). Building real time groupware with GroupKit, a groupware toolkit. *ACM Transactions on Computer Human Interaction*, **3**(1), pp. 66-106.

Shneiderman, B. (1997). *Designing the User Interface: Strategies for Effective Human-Computer Interaction.* 3nd edition, Ablex, Norwood, New Jersey.

Smith, R., Wolczko, M., and Ungar, D. (1997). From Kansas to Oz: Collaborative Debugging when a Shared World Breaks. *Communications of the ACM*, pp 72-78, April.

Sohlenkamp M. and Chwelos G. (1994). Integrating communication, cooperation and awareness: The DIVA virtual office environment. In *Proceedings of the ACM CSCW'94 Conference on Computer Supported Cooperative Work*, ACM Press.

Spellman, P., Mosier, J. and Deus, L. and Carlson, J. (1997). Collaborative Virtual Workspace. In *Proceedings of the Group'97 International Conference on Supporting Group Work*, pp. 197-203, ACM Press.

Sproull L. (1991). A lesson in electronic mail. In L. Sproull and S. Kiesler (Eds.), *Connections: New ways of working in the networked organization*. pp 177-184, MIT Press. Reprinted in Baecker (1993).

Sproull L. and Kiesler S. (1991). Increasing personal connections. In L. Sproull and S. Kiesler (Eds.), *Connections: New ways of working in the networked organization*, pp. 79-101. MIT Press. Reprinted in Baecker (1993).

Stefik, M., Foster, G., Bobrow, D., Kah,, K., Lanning, S. and Suchman, L. (1987a). Beyond the Chalkboard. *Communications of the ACM,* 30, 32-47.

Stefik M., Bobrow D. G., Foster G., Lanning S., and Tatar D. (1987b). WYSIWIS revised: Early experiences with multiuser interfaces. *ACM Transactions on Office Information Systems*, **5**(2), pp. 147-167, April.

Strauss A. (1993). *Continual permutations of action*, Aldine De Gruyter, New York.

Tang J. C. (1991). Findings from observational studies of collaborative work. In S. Greenberg, S. (Ed.), *Computer Supported Cooperative Work and Groupware*, pp. 11-28. Academic Press.

Tolone, W., Kaplan, S., and Fitzpatrick, G. (1995). Specifying dynamic support for collaborative work within wOrlds. In *Proceedings of the ACM COOCS'95 Conference on Organizational Computing*, ACM Press.

Turkle, S. (1995). *Life on the screen: Identity in the age of the Internet.* Simon and Schuster.

Turoff M. (1991). Computer-mediated communication requirements for group support. *Journal of Organizational Computing*, **1**, pp. 85-113. Excerpt reprinted in Baecker (1993).

Valacich, J., Dennis, A. and Nunamaker, J. (1991). Electronic meeting support: The GroupSystems concept. In S. Greenberg (editor), *Computer Supported Cooperative Work and Groupware.* Academic Press, p133-154, London, UK.

Wong, W. (1998). Team-building on the fly: Software lab report. *Byte*, pp. 106-110, February.

13. Appendix 1: Applets in TeamWave

Several of the applets included in TeamWave are described here. Most are fairly simply tools that can be used immediately with little or no training.

Applets for manipulating a visual workspace.

- *PostIt.* One useful applet is the ubiquitous sticky note (Figure 1e). Once created, any user in the room can change the text in the note at any time, with changes immediately seen on other users' displays.
- *Note Organizer.* The note organizer (Figure 1p) allows team members to compose text fragments on the applet window, and to move these fragments around. It is also a full groupware tool, where all people see the text as it is being edited, as well as seeing the items being repositioned.
- *Concept Map.* The concept map applet (Figure 1i) provides another way to organize information, this time as a graph. Nodes in the graph can represent ideas, while edges allow different ideas to be related. Users in the room can simultaneously add and edit nodes and edges. Nodes and edges are also typed. By default, the concept map is configured to the gIBIS methodology, where people can create nodes of the type issue, position, or argument, and link them together with edges of the type supports, objects-to, questions, is-suggested-by, and responds-to (Conklin 1988, Rein and Ellis 1991). Figure 1i illustrates a small concept map constructed using the gIBIS methodology. However, the group can reconfigure the concept map by creating their own set of node and edge types (Kremer 1997).
- *Calendar.* With the calendar tool (Figure 1k), people can create simple group schedules.
- *Games.* The tetrominoes puzzle (Figure 2e) is one of several "non-productivity" applets in TeamWave. The idea is to have several users simultaneously manipulate the colored pieces to fit them within the outlines. In a similar vein, TeamWave could provide applets such as a card table (no rules necessary, pick your own game with others in the room) or a chess board.
- *Database.* A very simple database applet is provided (Figure 1q). Users can define a set of fields, give each field a name and a type, and then manipulate records. While this is not a replacement for a full database, it is sufficient to hold small collections of information. Default configurations supplied with the system include an address book (Figure 1q) and a 'To Do' list.

Applets for navigating rooms.

- *Doorway.* People use the doorway applet to create a direct entrance to another room (Figure 1r). A group may use this, for example, as a quick way to direct other members of a group to relevant information found in a different room.

Applets that import or point to external information.

- *Image Tool.* Conventional rooms often contain graphical images, either work related or for decoration. We added an applet that could display an image in a room (Figure 1j). The image can be specified either by uploading an image file to the TeamWave server, or by specifying an URL which the applet uses to fetch the image from an HTTP server. Users can annotate and draw over the image using the whiteboard pens (Figure 1n).
- *File Transfer.* Another applet (Figure 1t) allows external files to be placed in rooms for later retrieval. By selecting the 'put' button, users can upload the file into the room, where it appears as an icon and filename. With the 'get' button, other users can later download the file from the room to their own file system.
- *External URL Reference.* To refer to external Web information from TeamWave, another applet holds a pointer to an URL. Clicking on the icon sends the URL associated with the applet to the

user's normal web browser, while clicking on the label allows changing the URL (Figure 1j). This is not a shared view application, for while the URL is shared, the action of invoking the browser is strictly local. People use this applet to leave pointers for other team members to look at on their own.

- *File Viewer.* The file viewer imports, uploads, and displays any ASCII file (Figure 2f+h). This is a relaxed 'what you see is what I see' display, where users can scroll independently to different parts of the file. A multi-user scrollbar indicates the portions of the file that other users can see (Figure 2g) (Roseman and Greenberg 1996c).
- *Web Browser.* As a final example of including outside information, the GroupWeb browser (Greenberg and Roseman 1996) was converted to a TeamWave applet (Figure 2d). This applet couples a simple Web browser with groupware features such as telepointers and page synchronization. Unlike the external URL reference applet, people can use this browser in real time to bring pages into a shared view, to navigate through those pages together, and to discuss information with others in real-time.

Custom-built applets.

- *Custom applets* can be created by programmers to fit particular group needs. They are provided as a "plug-in" to the TeamWave client, augmenting the built-in applets. Any community or third-party software developer can extend the applets on hand by creating their own suite of custom applets. Other possibilities include decision support tools, version control for external files, front-ends to external databases, and other information systems. These applets can be constructed reasonably by programmers using the GroupKit API (Roseman and Greenberg 1996c).