From Technically Possible to Socially Natural Groupware

Saul Greenberg and Carl Gutwin
Department of Computer Science
University of Calgary,
Calgary, Alberta Canada T2N 1N4
Tel: +1-403-220-6087
E-mail: saul@cpsc.ucalgary.ca

Cite as:

Greenberg, S. and Gutwin, C. (1998) **From Technically Possible to Socially Natural Groupware.** In *Proceedings of the 9th NEC Research Symposium: The Human-centric Multimedia Community*. August 31-September 1, Nara, Japan.

Abstract

Many people believe that computers can support small teams collaborating over distance in real time. On the surface, current technology now makes this goal possible. We have systems that afford communication (e.g., text chat, digital voice and video) and systems that afford sharable artifacts (e.g., electronic whiteboards, shared applications). While the technology is functional, the resulting systems are not. Quite simply, most real-time groupware does not yet afford a socially natural setting for learning, for work, and for play. The solution is that we have to pay far more attention to the human factors necessary for effective groupware. To illustrate the path towards "socially natural" groupware, I describe how my group has used human factors in the evolutionary design of a shared groupware workspace. In particular, I discuss how our designs grew from our understanding the human factors of: 1) tight coupling and its role in intense collaborations; 2) loose coupling and how it helps coordinate both group and individual work; 3) casual interaction and how it helps people contact one another; and 4) seamless transitions and how it influences the design of an integrated work environment.

Keywords: real time interaction, groupware, computer supported cooperative work, human factors.

Introduction

Multimedia technology is advancing at blinding rate. When applied to real time groupware, we now see demonstrations of systems that are technically impressive: shared displays; high bandwidth video and audio, populated virtual worlds, and so on. The problem is that many of these groupware systems do not pay attention to the often subtle human factors that form the nuances of everyday interpersonal interaction. Consequently, they are awkward to use, and they are rarely employed except by its creators, by early adapters, and by people whose needs are so pressing that they are willing to use even problematic systems.

The solution---and the message of this paper--is that:

> real time groupware must be designed around human factors before it can become a socially natural setting for learning, work, and play.

Fortunately, there is a large body of human factors research in groupware, arising mostly from the area of Computer Supported Cooperative Work (CSCW) (e.g., ACM CSCW Proceedings; Baecker 1993; Greenberg, Hayne and Rada 1995). Its researchers are trying to develop an intellectual foundation to groupware design in a variety of areas. Yet as a CSCW researcher, I am constantly surprised at how I (and others) have to revisit our design premises: we begin with what (in hindsight) is a fairly naive or incomplete hypotheses of the way people work, and as a consequence have to evolve our hypotheses and designs as the relevant human factors emerge.

In this paper, I will illustrate one path towards "socially natural" groupware, where I describe how my group has used human factors in the evolutionary design of a shared groupware workspace---systems that provide the equivalent of a tabletop or whiteboard that can be used by the group to collaborate over their artifacts in real time. Our goal over the years was to develop and/or apply existing human factors knowledge of face to face environments containing physical work surfaces to the design of workstation conferencing tools.

To tell this story, I will use the metaphor of a building (Figure 1). Its foundations are the technology that makes groupware possible by giving people the ability to communicate and share their computer artifacts. Its roof is socially natural groupware that lets people interact through their computers in a natural way. The pillars that support this roof were built from our understanding the human factors of:

- **tight coupling** and its role in intense collaborations;
- **loose coupling** and how it helps coordinate both group and individual work;
- **casual interaction** and how it helps people contact one another; and
- **seamless transitions** and how it influences the design of an integrated work environment.

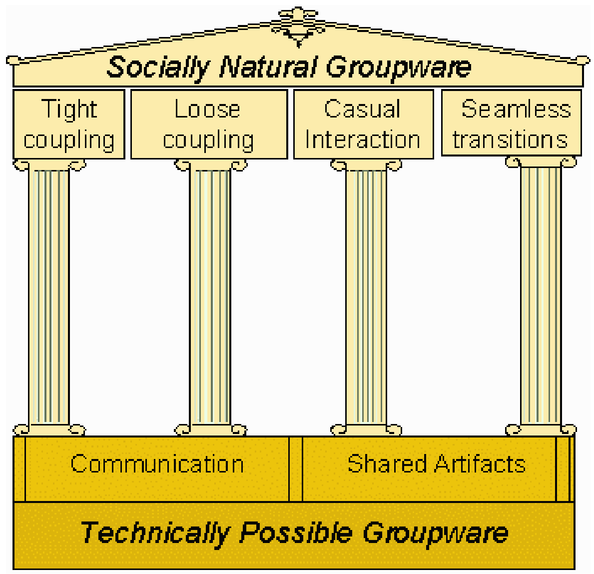

Figure 1. A 'building' illustrating how we moved from technically possible to socially natural groupware workspaces

Technically Possible Groupware Workspaces

Shared groupware workspaces have been around for quite a while (Engelbart and English, 1968), both as *shared view systems* and *groupware drawing editors.*

The first form is the shared view system, where each distant-separated participant would have an identical view of the running single-user application and an opportunity to interact with it. The variety of papers describing these systems (e.g., Greenberg 1990; Lauwers, Lantz, and Romanow1990; Ensor et.al, 1988) pay particular attention to the technical challenges of building them. While shared view systems are useful (since they are sometimes the *only* way to share existing computer artifacts over distance), these systems had severe drawbacks. Because the underlying application had no knowledge that multiple people were using it, people had to work serially by taking turns. This led to unwieldy add-on turn-taking interfaces (Greenberg 1991), or "cursor wars" where people would fight for who was controlling the cursor and text insert spot. Another problem is that these systems only allow people to view the same thing: there was no way, for example, for individuals to scroll to a different part to the work area without affecting the display of others. Similarly, people cannot customize the application to support individual working styles.

In the mid-1980s, a variety of researchers developed true groupware systems that recognized that more than one person could be using them (e.g., Lee, 1990; Stefik et. al., 1989; Foster and Stefik 1986). Most were shared drawing editors, where people could draw something on their display, and where other distant people could see the drawing. Technical concerns often motivated these designs. In XSketch (Lee 1990), for example, a person's drawing act was not transmitted until after the act was complete, ostensibly to save network transmissions. That is, if someone drew a line, the other person would see nothing until after the entire drawing stroke was completed. In We-Met, people could see the unfolding drawing, but had no way to point to things (Wolf, Rhyne, and Briggs), again to save either bandwidth or processing power. Several systems also had the expectation that people would communicate via typing, reducing the need for video and audio transmissions. Figure 2 illustrates such a prototypical system, where the person on the left is in the act of drawing a rectangle. The person on the right does not yet see this rectangle (since it is still in the rubber band state), nor can they see the other person's pointer. Typing is through the chat box at the bottom and is, of course, an unwieldy way to communicate. While the resulting state of the workspace will (eventually) be identical, it does not afford easy communication over the work being done on it (see next section). Tang (1991) summed the problem up nicely when he said that designers of these systems based it on the naive notion that shared worksurface were simply for storing artifacts. As we will see in subsequent sections, the fact that these systems worked from a technical point of view does not imply that they are usable.

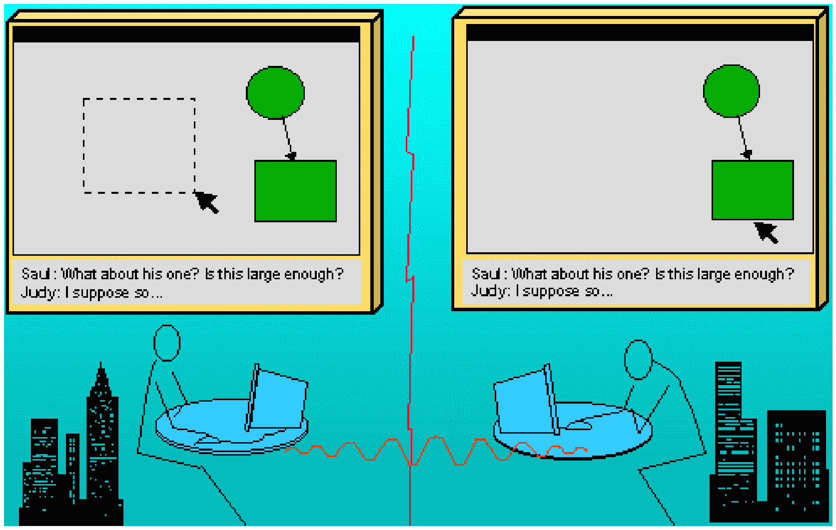

Figure 2. A prototypical early system. Intermediate states of objects are not transmitted, other people's pointers are not visible, and people are expected to communicate by typing.

Tight Coupling: The Human Factors of Intensive Collaborations

Our first pillar supporting our roof of socially natural groupware (Figure 1) considers the human factors of tight coupling, where people are intensely collaborating over a shared work surface.

As mentioned, a naive view of the communal work surface would consider it merely as a medium for creating and storing a drawing artifact (Tang 1991). Bly (1988) disproves this naive view. She studied two designers communicating through three different media offering different access to a drawing surface: face to face including a shared sketchpad; over a video link that included a view of the other person and their personal drawing surface; and over the telephone. From her observations, she asserts that the drawing process—the actions, uses, and interactions on the drawing surface—are as important to the effectiveness of the collaboration as the final artifact produced. Bly also noticed that allowing designers to share drawing space activities increases their attention and involvement in the design task. When interaction over the drawing surface is reduced, the quality of the collaboration decreases. In effect, the artifacts in that space act as stage and props for rich person-to-person interaction.

Tang refined Bly’s findings even further through his ethnographic study of eight short small-team design sessions (Tang 1991; also see Garner, Scrivner, Clarke et al, 1991 for further work in this area). Each team used large sheets of paper as a shared work surface and were given problems to solve. From his observations, Tang built a descriptive framework to help organize the study of work surface activity, where every user activity was categorized according to what action and function it accomplished, as listed below.

<u>Actions:</u>

- listing produces alpha-numeric notes that are spatially independent of the drawing;
- drawing produces graphical objects, typically a 2-dimensional sketch with textual annotations that are attached to the graphic;
- gesturing is a purposeful body movement that communicates specific information, e.g. pointing to an existing drawing.

<u>Functions:</u>

- storing information refers to preserving group information in some form for later recall;
- expressing ideas involves interactively creating representations of ideas in some tangible form, usually to encourage a group response;
- mediating interaction facilitates the collaboration of the group, and includes turn-taking and focusing attention.

Tang’s classification of small group activities within this framework revealed that the "naive" view of work surface activity—storing information by listing and drawing—constitutes only ~25% of all work surface activities. Expressing ideas and mediating interaction comprised the additional ~50% and ~25% respectively. Gesturing, which is often overlooked as a work surface activity, played a prominent role in all work surface actions (~35% of all actions). For example, participants enacted ideas by using gestures to express them, and gestures were used to signal turn-taking and to focus the attention of the group. He also noticed that, given good proximity, a high percentage (45–68%) of work surface activity around the tabletop involved simultaneous access to the space by more than one person. These and other observations led him to derive six design criteria that shared work surface tools should support (Table 1). He stresses the importance of allowing people to gesture to each other over the work surface, and emphasizes that the process of creating a drawing is in itself a gesture that must be shown to all participants through continuous, fine-grained feedback. Another key point is that the tool must not only support simultaneous activity, but also encourage it by giving participants a common view of the work surface.

Design Criteria	Reasons (Human Factors)
1) Provide ways of conveying and supporting gestural communication.	• gestures are a prominent action • gestures are typically made in relation to objects on the work surface • gestures must be seen if they are to be useful • gestures are often accompanied by verbal explanation
2) Minimize the overhead encountered when storing information.	• only one person usually records information • other participants should not be blocked from continuing private or group work while information is being stored
3) Convey the process of creating artifacts to express ideas.	• the process of creation is in itself a gesture that communicates information • speech is closely synchronized with the creation process • artifacts in themselves are often meaningless
4) Allow seamless intermixing of work surface actions and functions	• a single action often combines aspects of listing, drawing and gesturing • writing and drawing alternates rapidly • actions often address several functions
5) Enable all participants to share a common view of the work surface while providing simultaneous access and a sense of close proximity to it.	• people do not see the same things when orientation differs • simultaneous activity is prevalent • close proximity to the work surface encourages simultaneous activity
6) Facilitate the participants’ natural abilities to coordinate their collaborations	• people are skilled at coordinating communication • we do not understand the coordinating process well enough to mechanize it

Table 1. Six criteria for designing a communal work surface (condensed from Tang 1989)

Several systems were then developed from these design principles by a variety of researchers that tried to support the kinds of tightly coupled interactions people expect from a group drawing surface. While these systems are quite diverse, they all share a common feel, and observations of use are strikingly similar. Our own early systems, GroupSketch and GroupDraw (Greenberg, Roseman, Webster and Bohnet 1992) were workstation-based multi-user sketch and drawing applications. As partially illustrated by the GroupSketch interface in Figure 3, we followed Tang's guidelines by: giving people explicit identical views; making all fine-grained actions immediately visible to all people; providing telepointers for gesturing; allowing simultaneous activity; and by not imposing any social protocol on the group. In parallel, Bly and Minneman (1990) developed Commune: its main difference from GroupSketch is that people used a stylus to write directly on top of the horizontally-oriented monitor while ours required people to draw (awkwardly) with a mouse. Other systems developed were video-based and work by fusing video images: Early examples are VideoDraw (Tang and Minneman 1990), VideoWhiteboard (Tang and Minneman 1991) and TeamWorkStation (Ishii 1990).

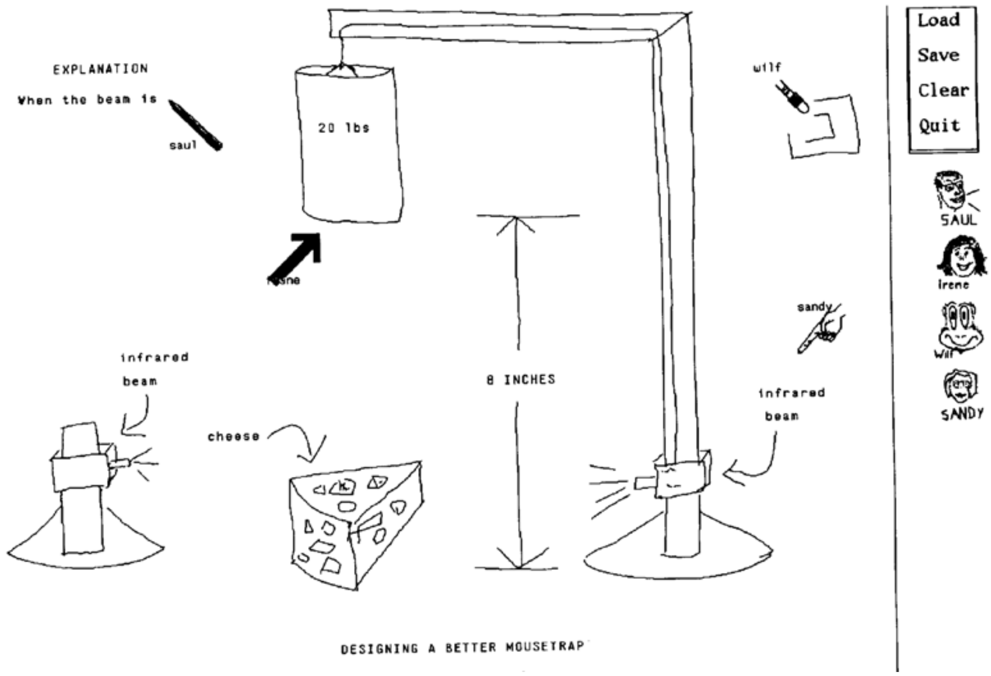

Figure 3. A sample GroupSketch session

Loose Coupling: The Human Factors of WorkSpace Awareness

While we were initially satisfied with GroupSketch and its descendants, we eventually realized that our designs were too limited for serious use. The problem is that the notion of tight coupling was too restrictive. What we had overlooked was that in prolonged interaction, group members shift their attention back and forth between individual and shared activities. Yet many of these early systems did not support individual activities particularly well. Some did not allow people to scroll independently, which meant that individuals could not go to a different part of the work surface without affecting other participants. Other systems did allow independent scrolling (Stefik, Bobrow et. al. 1987), but suffered the problem that people could then no longer know what others were doing. What we needed to do was to understand the human factors of loose coupling (i.e, when people perform individual activities within a communal setting) and how people moved between loose and tight coupling. This is the second pillar in Figure 1.

When people work together in a face-to-face setting, a wide variety of perceptual cues help them track of what others are doing. People notice activities of others by hearing sound events and through their peripheral vision; they can glance around the work surface to get the big picture; they can quickly look up and refocus on another area to see the details of what another person is doing. People will occassionally verbally shadow their actions to inform others i.e., they say what they are doing as they are doing it. This awareness of others in the workspace is what we call *workspace awareness*, the up–to–the–moment understanding of another person's interaction with the shared space (Gutwin, Greenberg and Roseman, 1996a and 1996b; Gutwin and Greenberg 1998a). At a simple level, it involves knowledge of who is present, where they are working, and what they are doing. Workspace awareness is critical to effective collaboration, as it helps coordinate activity, simplifies verbal communication, and manages movement between individual and shared work.

Unfortunately, current groupware systems provide only a fraction of the information needed to maintain workspace awareness as they lack many of the natural affordances that exists in face-to-face settings. For example, consider the typical solution providing for individual work by allowing independent scrolling. In Figure 4, we show a large work area, where two people can see only part of this on their display. This is represented at the interface level in Figure 5, where we see exactly what is on each person's display. Because each sees completely different things, there is no visible frame of reference between them and workspace awareness is lost. They don't know where the others are in the workspace and they cannot tell what they are doing. Any gestures made are invisible. Indeed, it is extremely difficult to gain a sense of the 'big picture' of what is going on.

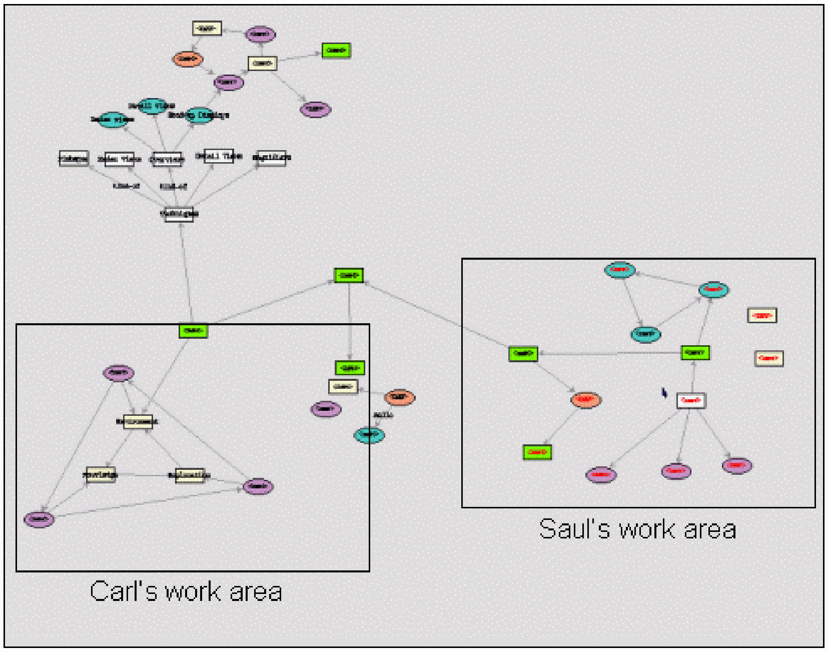

Figure 4. The work area and two people's viewports into it

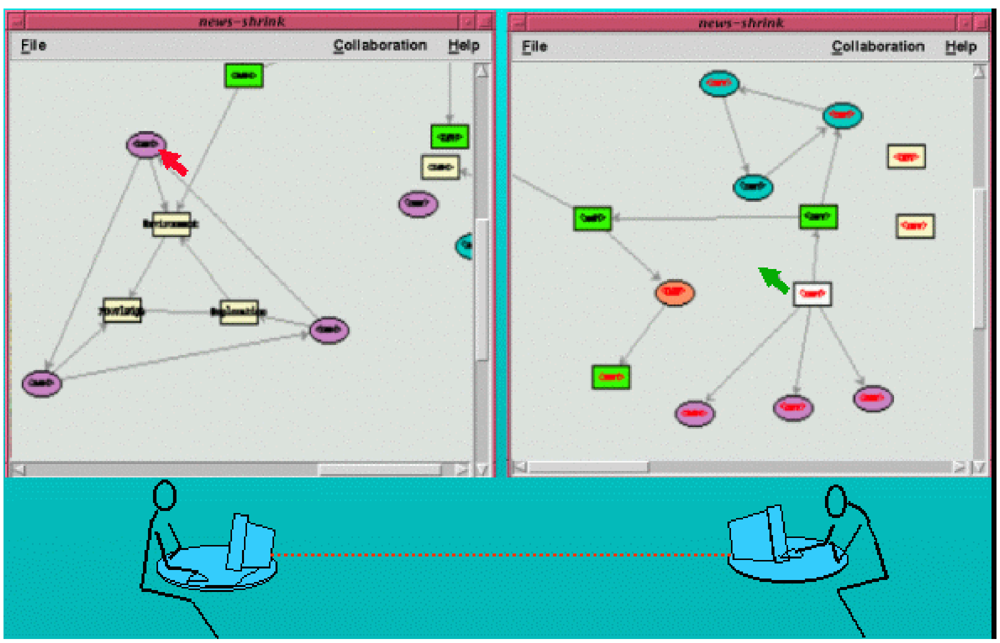

Figure 5. What the two people can see when they are located in different parts of the work surface

Within this strange new situation, the groupware designer must try and recreate the conditions and cues that allow people to keep up a sense of workspace awareness. Whereas face-to-face interaction has inherent mechanisms and affordances for maintaining workspace awareness, the groupware designer is faced with a blank slate---any support for building or maintaining workspace awareness must be explicitly determined and built into the groupware system.

Consequently, we developed a wide variety of visualization techniques that supplies collaborators with the workspace awareness they require for managing their loose and tight couplings (Gutwin, Greenberg and Roseman 1996a; Gutwin and Greenberg 1998c). These include radar overviews (Gutwin and Greenberg 1998b), fisheye views (Greenberg, 1996a; Gutwin and Greenberg 1998c), and transparent layers (Gutwin and Greenberg 1998c). We will use the radar overview as an example (Figure 6). The radar overview provides a miniature of the entire workspace as part of the interface (the upper left corner of Figure 6). Onto this we overlay the view extents of what others can see, inlaid with that person's image. Telepointers are also visible within the radar overview as well as in the detailed main view. Since the radar overview provides a spatial representation of the workspace as well as the state of items within it, it is easy to see who is present, where others are located, and what others are doing.Telepointers within the overview let people gesture around as well. The radar overview is also active, which means that people can work in both the overview or the detailed view. Finally, people can grab their view extents rectangles in the overview and move them. In turn, this means that people can quickly do the equivalent of a detailed glance at another's work simply by moving their view rectangle atop the others, which will bring their detailed views into alignment. We have run extensive studies on this approach, and the results suggest that these systems are effective at providing people with the workspace awareness necessary for managing loose and tight coupling (Gutwin and Greenberg 1998b).

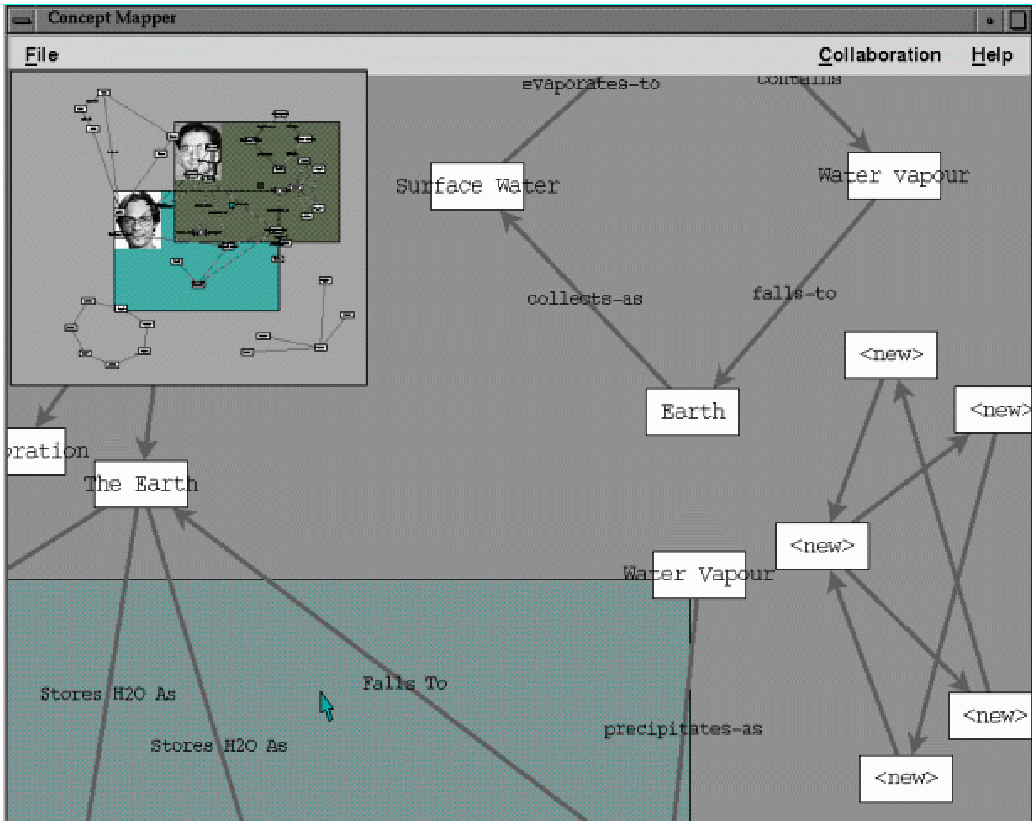

Figure 6. The Radar Overview

Casual Interaction: Supporting the Human Factors of Lightweight Contact

At this point, we were comfortable with the software we had designed. Yet we faced a difficult and somewhat embarrassing problem: we did not use it. Some of this failure, of course, reflects both the functional limitations inherent with proof of concept systems and demonstrable prototypes. However, we realized that even if we overcame these limitations, we would use the system infrequently simply because the effort required to get in touch with another person and to start up the software was onerous. What we had ignored was the human factors of casual interaction, and how people move from awareness of others to light-weight communication and work. This is a third pillar of Figure 1.

In a landmark study, Kraut, Egido and Galegher (1988) studied how people meet. They noticed that the backbone of everyday coordination and work between co-located team members is casual interaction, the spontaneous and one-person initiated meetings that occur over the course of the day . The glue behind these interactions is informal awareness, where people track and maintain a general sense of who is around and what others are up to as they work and mingle in the same physical environment (Cockburn and Greenberg 1993). People subconsciously ask and answer questions such as:

- who is around?
- are they available now?
- can they be interrupted?
- how can I initiate contact with them?

In co-located settings, this is easy to do. We bump into one another, we briefly exchange greetings and information, we carry our conversations down hallways giving others the opportunity to join us; we move into work by using ready to hand materials such as napkins and the backs of envelopes.

In contrast, casual interaction in distributed communities is difficult, as people do not have the same opportunities to interact with one another. Indeed, most software is designed around the notion of a formal meeting, where we somehow establishing a 'call' or 'session' between interested parties. This is heavyweight. For example, Figure 7 illustrates what one would have to do in the Netscape Conference system to establish contact with another person. First, they have to determine how to get in touch with the other person, which includes knowing their email address and entering it (or selecting it from a phonebook). Next they have to initiate the call. This usually fails, either because the other person does not have the software running or because they are not there (computer tag is far worse than telephone tag!). If this does work, they then have to establish the communication channel, then bring in the applications they want, and finally load any information into the application if they wanted to refer to something from a previous session. While reasonable for formal meetings, this is completely inappropriate for the light-weight casual interactions mentioned above.

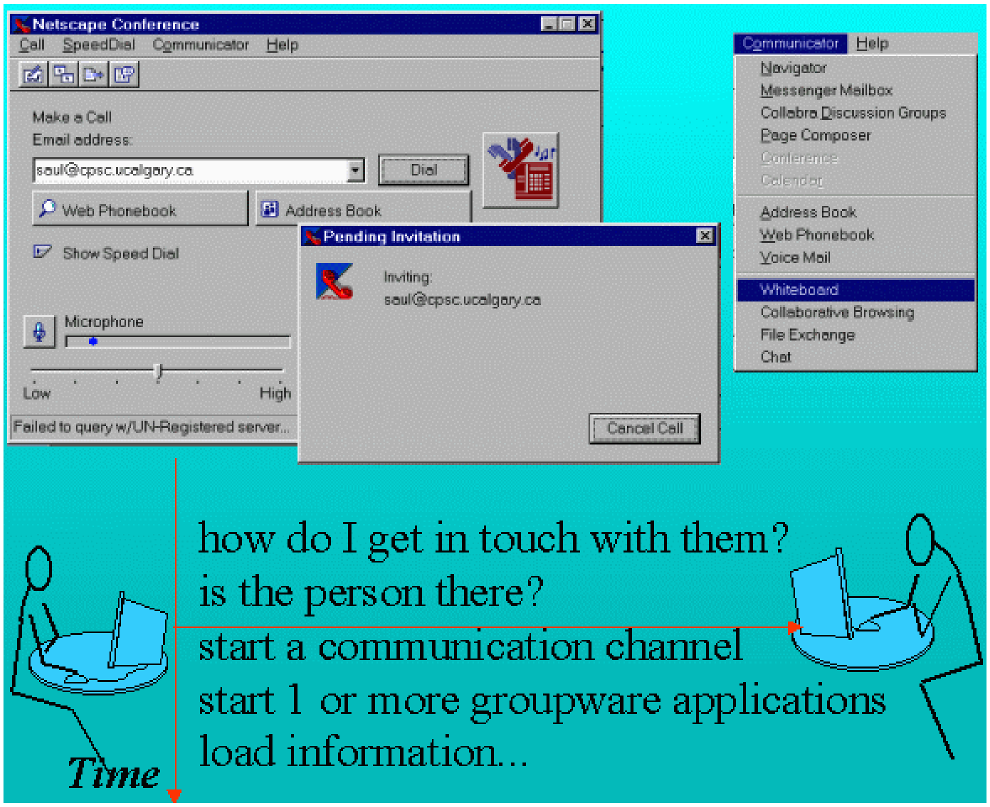

Figure 7. The heavy-weight establishment of communication in Netscape Conference

Consequently, CSCW researchers have developed a variety of methods for providing informal awareness information and for mediating casual interaction in distributed communities. Through media spaces, users select offices and common areas at remote sites and view them through continuous video (Bly, Harrison, and Irwin 1993), or by brief glances into selected offices (Tang, Isaacs, and Rua 1994). A low bandwidth alternative uses periodic video snapshots instead of a video stream (Dourish and Bly 1992), where the community is presented on one's screen as an array of small images updated few minutes. Another approach is iconic presence indicators, which use stylized icons to show who is around and the likelihood of their availability e.g., Peepholes (Greenberg 1996b) and Red Light-Green Light (Wax 1996). Unlike video systems that simply transmit a snapshot of the real physical situation, iconic indicators are based on abstractions of awareness information. For example, most simply monitor a person's computer activity, and use the idle time between activities to estimate a person's presence.The advantage is that these iconic systems try to present *only* enough information for awareness and casual interaction, which mitigates concerns about using this technology for surveillance and privacy intrusions.

We will use our own Peephole iconic indicator (Greenberg 1996b) system to illustrate how this works (Figure 8). Each icon knows how long it has been since particular people was active on their computer, and uses it to display an estimate of that person's real availability. For example, we see that Greenberg is now active and likely available (denoted by a bold character), O'Grady has been idle for a few minutes (the grayed out icon), Lowe is logged on but hasn't used the computer in a while and is probably unavailable, Schaffer is logged off, and Roseman is unreachable. A quick glance at these icons gives awareness of people's probable availability for real time communication. While these activity indicators suggest when a call will work, they must be monitored regularly to see when an absent person returns. Indeed, users of the Cruiser media space would often open a full bandwidth video connection to the empty office of a collaborator, solely to 'ambush' its occupant, i.e., to see when they returned (Fish, Kraut, Root and Rice 1992) . In Peepholes, users can ambush others through a menu option (Figure 8). When the system notices that the person has become active, it announces their return by playing an audible sound of someone typing. This attracts the ambusher's attention to the display, allowing them to initiate a call if desired. Peepholes also allows people to move from awareness of another's availability to an informal meeting: groupware connections are established by simply selecting a person's Peephole icon and an application icon (Figure 8, bottom), and the applications will be started. Connections are literally a button click away.

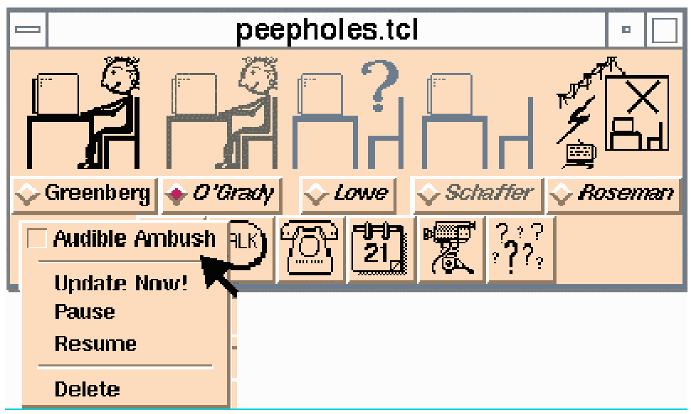

Figure 8. Peepholes

With higher end technology, we can produce even more interesting systems that mitigate privacy while affording casual awareness and interaction. Figure 9 illustrates an approach that we have been taking with researcher Hideaki Kuzuoka from Tsukuba University (Kuzuoka and Greenberg 1998). The unit portrayed in the figure, called the *Active Hydra,* embodies a video/ audio connection to a single remote person. We instrumented Hydra units (Buxton 1997; Buxton, Sellen and Sheasby 1997) with proximity sensors that measure how close a person is to it. Unlike the original Hydra, the presence or absence of the audio, the quality of the video, and the presence of groupware on the computer display is controlled implicitly by a person’s position relative to the Hydra unit. When both people are close to their Hydra surrogates, they have a full audio/video channel, and groupware on the computer is activated (e.g., a shared sketchpad is made ready-to-hand to augment communication). As one moves away from the surrogate, audio is disabled. Moving even further away degrades the video to occasional glimpses into each other’s space, and the groupware on the computer disappears. As well, people can explicitly control how permeable this system is by the position of a doll surrogate (foreground): if it is on the podium, it means that we are making ourself very available to the other person. Off the podium expresses some availability, but restricts the channel somewhat. Tipping the doll over indicates minimal availability. For redundancy, the doll in the background represents the activity level of the other person. If that person is present in their office, the doll faces forward. If they leave, it turns to face the wall (Kuzuoka and Greenberg 1998).

Figure 9. The Active Hydra Unit

Seamless Transitions: The Human Factors Behind an Integrated Environment

At this point in our development, we had a fairly good idea of the human factors necessary for creating effective groupware work surfaces, as well as how we coulld get people together to actually work on them. However, there were still some elements missing. In particular, we realized that our designs were still too constrained, and they did not account for the variety of things that people did during their working day. For example, our systems focused only on groups doing real time interaction. We conveniently ignored the fact that in the real world, people move themselves and their artifacts continually and effortlessly between different styles of collaboration: across time (real time and asynchronous), across individual and group activity, across place, across formality, and so on. The problem was that our systems --- like many others --- had gaps that made it difficult or even impossible for people to make the transition from one collaboration style to another (Baecker 1993). To move across these gaps, people had to make fairly heavy-weight and disruptive transitions within and between software. Alternatively, they would decide that the personal cost was too great and do without groupware support. In essence, the problem was that we had to design systems to support seamless transitions between working styles, which is the fourth pillar in Figure 1.

Our approach to solving this problem was to consider how existing real world environments naturally provided ways to support different working styles. In particular, we wanted to see how the properties of dedicate team and project rooms could be brought into the electronic forum (Covi, Olson and Rocco 1998). These rooms are open offices housing a team of 3-6 people, and provide a shared space where people can work together and leave their artifacts. They naturally support seamless interaction (Figure 10) (Greenberg and Roseman 1998). They supply:

- a place for both individual *and* group work
- a place for both formal and informal face to face meetings
- a place to leave reminders, notes and work artifacts for others (i.e., asynchronous interaction)
- a place that supplies many opportunities for light-weight casual interaction

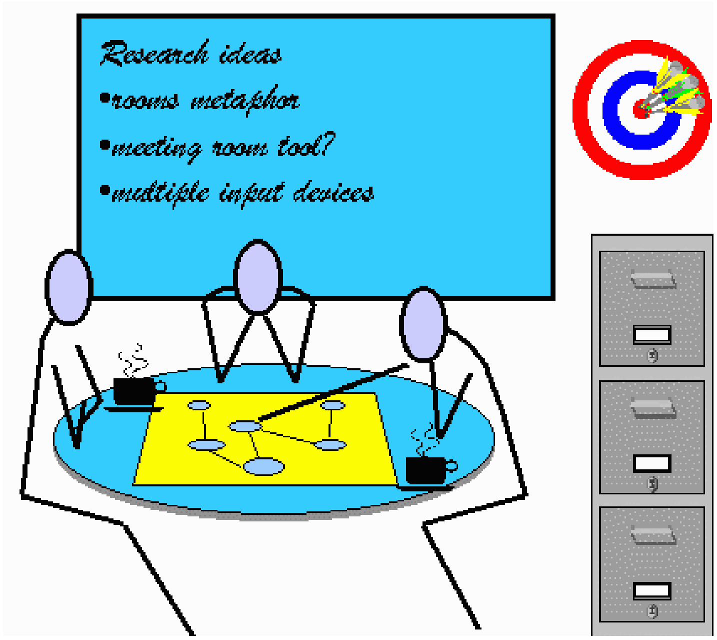

Figure 10. A stylized team room

We developed a system called TeamRooms (Roseman and Greenberg 1996), later commercialized as Teamwave Workplace (http://www.teamwave.com), that was loosely based on the idea of a team room metaphor. The best way to illustrate some of its properties is through a scenario that demonstrates (in principle) how people can make light-weight transitions between different styles of work. Greenberg and Roseman (1998) provides a complete description of these ideas.

The scenario considers how a team prepares and pursues a committee meeting. Dubs and Hayne (1992) see this style of meeting as a process that cycles through three generic phases:

- pre-meeting setup: setting goals, getting participants, collecting materials;
- during-meeting activities where people work together face to face to pursue their work;
- post-meeting tear-down, where documents are created and distributed to others (which could also lead into the next meeting).

Let us begin with the pre-meeting, and show how TeamRooms affords individual work, casual interaction, and asynchronous interaction. We will use the screen snapshot in Figure 11, which shows the state of one room at the start of the during meeting process, and we will describe how it arrived at that state from the pre-meeting process. Saul is charged with setting up the first meeting for this team, and begins his individual work by creating a room called "Meeting Room". Using a special purpose roster applet (top left sub-window), he then jots down the meeting goal and the roster of potential attendees. With the note organizer, he notes agenda points (middle left). He indicates the time of this and subsequent meetings on the calendar (upper right). 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 Budget) so that others can retrieve and read it ahead of time. 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 (the yellow note, middle right).

Other people then drop into the room to see what is there (this is a form of asynchronous interaction, as Saul has left these items as well as notes for others to review). Judy simply tells others, through the agenda tool, that she will be absent. A few hours later, Carl enters, adds a fourth agenda item, and includes a URL pointer to a document that he has worked on that is relevant to this point. Mark enters the room shortly after, sees Carl, and they start chatting (this is an example of casual interaction). Together, they decide to bring in 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. Mark wonders if George should be allowed in, but Carl says that others may object. Because they are not sure, they include a voting tool which asks this question. Later, Carl comes back into this room and continues working on the document attached to agenda item 4.

When the meeting actually begins, participants see one another "walk through the door" as their icons become visible on the narrow panel on the left and as their 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, and new rooms can be created as required. Salient meeting points can be easily recorded, perhaps by inserting notes into the room, and by adding action items to the action item list.

The post-meeting process, which is mostly asynchronous an individual work, is straight-forward as well. The state of the room becomes part of the meeting record, and a 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 at this point becomes a medium for the group to communicate asynchronously to each other.

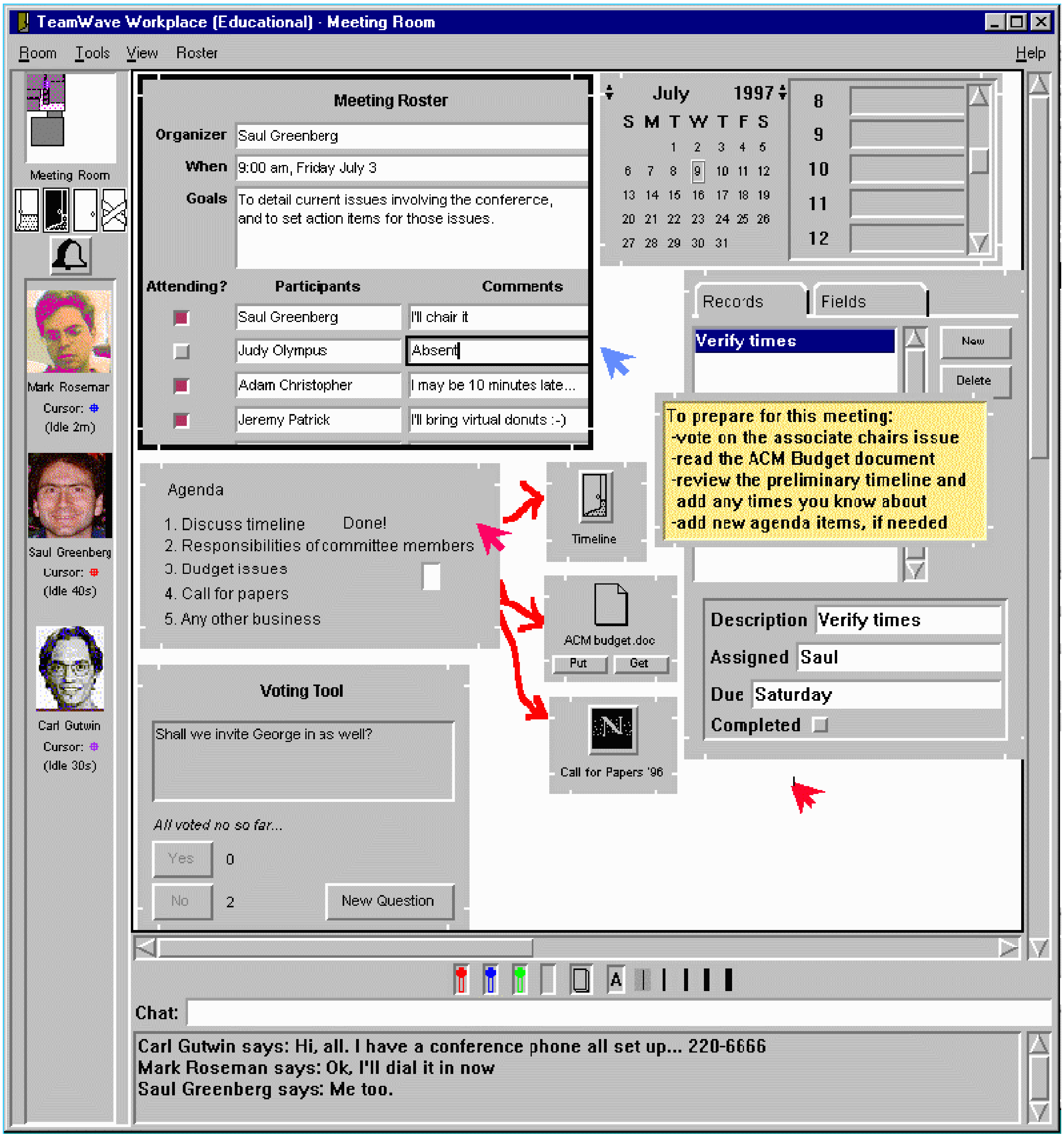

Figure 11. A TeamWave room. At the time of this snapshot, people have gathered together for a scheduled meeting

Conclusions.

I began this paper by claiming that many groupware systems are awkward to use. I suggested that real time groupware must be designed around human factors before it can become a socially natural setting for learning, work, and play. I described how my group has used human factors in the evolutionary design of a shared groupware workspace. In particular, I discussed how our designs over the last ten years grew from our understanding the human factors of: 1) tight coupling and its role in intense collaborations; 2) loose coupling and how it helps coordinate both group and individual work; 3) casual interaction and how it helps people contact one another; and 4) seamless transitions and how it influences the design of an integrated work environment. It should be clear from these examples that the systems we developed are a far cry from the early "technically possible" groupware, and that they do a better job --- at least in principle --- of providing people with a more natural setting for their work and play.

Of course, these systems have a long ways to go. It will take many more knowledge of human factors and how it can be applied effectively to groupware if we are to support this roof of "socially natural groupware". At best, we have a roof that is in a delicate balance: this is illustrated in Figure 12 which shows a "side view" of Figure 1. The point is that while some things are now natural, others are still awkward. Our roof still needs more pillars of human factors knowledge.

As multimedia and groupware developers, we must take all this into account. We must know the existing body of human factors and how it applies to our designs. We must understand the limitations in our knowledge, as some of the necessary human factors foundations are still being formed.We must be pro-active and discover what new human factors are relevant to our audience.

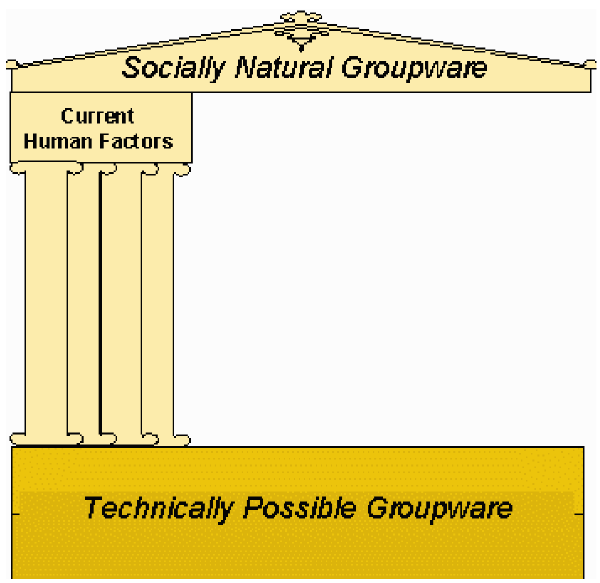

Figure 12. Side View: The delicate balance

Acknowledgements.

This paper reflects work done over the years by many members and affiliates of GroupLab, including Mark Roseman, Carl Gutwin, Ted O'Grady, Hideaki Kuzuoia, Andy Cockburn, Ralph Bohnet and David Webster. The work has been sponsored by various agencies, including the National Science and Engineering Research Council of Canada (NSERC), and Intel Corporation.

References

1. Baecker R. (1993). **Readings in groupware and computer-supported cooperative work: Assisting human-human collaboration.** Morgan-Kaufmann, San Mateo, CA.
2. Bly, S. (1988) **A use of drawing surfaces in different collaborative settings.** In *Proceedings of the Conference on Computer-Supported Cooperative Work (CSCW '88)*, pp. 250-256, Portland, Oregon, September 26-28, ACM Press.
3. Bly, S., Harrison, S. & Irwin, S. (1993) **Media spaces: Bringing people together in a video, audio and computing environment.** *Comm ACM*, 36(1), 28-47.
4. Bly, S. A. and Minneman, S. L. (1990) **Commune: A shared drawing surface.** In *Proceedings of the Conference on*

Office Information Systems, pp. 184-192, Boston, April 25-27.
5. Buxton, W. (1997). **Living in augmented reality: Ubiquitous media and reactive environments.** In Finn, A. Sellen, A. & Wilber, S. (Eds.). *Video Mediated Communication*. Hillsdale, N.J.: Erlbaum.
6. Buxton, W., Sellen, A. & Sheasby, M. (1997). **Interfaces for multiparty videoconferencing.** In Finn, A. Sellen, A. & Wilber, S. (Eds.). *Video Mediated Communication*. Hillsdale, N.J.: Erlbaum.
7. Cockburn, A. & Greenberg, S. (1993). **Making contact: Getting the group communicating with groupware.** *Proc ACM Conference on Organizational Computing System*.
8. 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.
9. 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
10. Dourish P. & Bly S. (1992). **Portholes: Supporting awareness in a distributed work group.** *Proc ACM CHI'92*.
11. Engelbart, D. and English, W. (1968) **A research center for augmenting human intellect.** *Proceedings of Fall Joint Computing Conference*, 33(1), p395-410, AFIPS Press.
12. Ensor, J.R., Ahuja, S.R., Horn, D.N. and Lucco, S.E. (1988) **The RAPPORT multimedia conferencing system — a software overview.** In *Proceedings of the IEEE Conference on Computer Workstations*, p52–58, March.
13. Fish R., Kraut R., Root R. and Rice R. (1992) **Evaluating video as a technology for informal communication.** *Proc ACM Conference on Human Factors in Computing Systems*, pp. 37-48, May 3-7.
14. Foster, G. and Stefik, M. (1986) **Cognoter, Theory and practice of a Colab-orative tool.** In *Proceedings of the Conference on Computer-Supported Cooperative Work*, pages 7-15, Austin, Texas, December 3-5.
15. Garner, S. W., Scrivner, S. A. R., Clarke, A. A., Clark, S., Connolly, J. H., Palmen, H., Schappo, A. and Smyth, M. G. (1991) **The use of design activity for research into computer supported co-operative work (CSCW)**. In *Proceedings of DATER '91*, pp. 84-96, Loughborough, UK, September.
16. Greenberg, S. (1996a) **A Fisheye Text Editor for Relaxed-WYSIWIS Groupware**, *ACM SIGCHI'96 Conference on Human Factors in Computing System, Companion Proceedings*, p212-213.
17. Greenberg, S. (1996b) **Peepholes: Low cost awareness of one's community**, *ACM CHI'96 Companion.*
18. Greenberg, S. (1991). **Personalizable groupware: Accomodating individual roles and group differences**. *Proceedings of the ECSCW '91 European Conference of Computer Supported Cooperative Work*, p17-32, Amsterdam, Kluwer Academic Press.
19. Greenberg, S. (1990). **Sharing views and interactions with single-user applications.** In *Proceedings of the ACM/IEEE Conference on Office Information Systems*, p227-237, Cambridge, Massachusets.
20. 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. ISBN 0-07-707899-3.
21. 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. http://cpsc.ucalgary.ca/grouplab/papers/
22. Greenberg, S., Roseman, M., Webster, D. and Bohnet, R. (1992). **Human and technical factors of distributed group drawing tools.** *Interacting with Computers*, 4(1), p364-392. Butterworth-Heinemann.
23. Gutwin, C. and Greenberg, S. (1998a) **Design for Individuals, Design for Groups: Tradeoffs between power and workspace awareness.** *Proceedings of the ACM Conference on Computer Supported Cooperative Work.* ACM Press.
24. Gutwin, C. and Greenberg, S (1998b). **Effects of Awareness Support on Groupware Usability.** *Proceedings of the CHI'98 Conference on Human Factors in Computing Systems*, ACM Press.
25. Gutwin, C. and Greenberg, S. (1998c). **Focus and Awareness in Groupware**. *Video Proceedings of the ACM Conference on Computer Supported Cooperative Work.* ACM Press. Videotape and abstract.
26. Gutwin, C., Greenberg, S. and Roseman, M. (1996a). **Workspace Awareness in Real-Time Distributed Groupware: Framework, Widgets, and Evaluation**. in Sasse, R.J., A. Cunningham, and R. Winder, Editors. *People and Computers XI (Proceedings of the HCI'96)*, pages 281-298, Springer-Verlag. Conference held at Imperial College, London, August 20-23.
27. Gutwin, C., Greenberg, S., Roseman, R. (1996b) **Supporting Awareness of Others in Groupware.** A short paper suite, in *ACM SIGCHI'96 Conference on Human Factors in Computing System, Companion Proceedings*, p205-215.
28. 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.
29. Kraut, R., Egido, C. & Galegher, J. (1988) Patterns of contact and communication in scientific collaboration. *Proc ACM CSCW'88.*
30. Kuzuoka, H. and Greenberg, S. (1998) **Mediating Awareness and Communication through Digital but Physical Surrogates.** Research Report 98-631-22, Department of Computer Science, University of Calgary, Calgary, Alberta,

Canada. Includes a companion 7 minute video of the same title
31. Lauwers, J.C., Lantz, K.A. and Romanow, A.L. (1990) **Replicated architectures for shared window systems: A critique.** *In COIS 90—Proceeding of the Conference on Office Information Systems*, Cambridge, Mass. April 25-27.
32. Lee, J. (1990). **XSketch: A multi-user sketching tool for X11**. In *COIS 90—Proceeding of the Conference on Office Information Systems*, Cambridge, Mass. April 25-27, p. 169-173.
33. Roseman M. and Greenberg S. (1996). **TeamRooms: Network places for collaboration.** In *Proceedings of ACM CSCW'96 Conference on Computer Supported Cooperative Work*, Boston, USA, ACM Press.
34. Stefik, M., Foster, G., Bobrow, D., Kahn, K., Lanning, S. and Suchman, L. (1987) **Beyond the chalkboard: Computer support for collaboration and problem solving in meetings.** Communications of the ACM, 30(1), p32-47.
35. Stefik, M., Bobrow, D. G., Foster, G., Lanning, S. and Tatar, D. (1987) **WYSIWIS revised: Early experiences with multiuser interfaces.** *ACM Trans Office Information Systems*, 5(2), pp. 147-167, April.
36. Tang, J. C. (1991) **Findings from observational studies of collaborative work**. In *Computer Supported Cooperative Work and Groupware,* S. Greenberg ed. Academic Press. Also in Int J Man Machine Studies, 34(2), p143-160, February.
37. Tang, J. C. and Minneman, S. L. (1990) **Videodraw: A video interface for collaborative drawing**. In *ACM SIGCHI Conference on Human Factors in Computing Systems*, pp. 313-320, Seattle Washington, April 1-5, ACM Press.
38. Tang, J. C. and Minneman, S. L. (1991) **VideoWhiteboard: Video shadows to support remote collaboration.** In *ACM SIGCHI Conference on Human Factors in Computing Systems*, pp. 315-322, New Orleans, April 28-May 2, ACM Press.
39. Tang, J., Isaacs, E. & Rua, M. (1994) **Supporting distributed groups with a montage of lightweight interactions.** *Proc ACM CSCW'94.*
40. Wax, T. (1996) **Red light, green light: Using peripheral awareness of availability to improve the timing of spontaneous communication**. In *Short Paper Proceedings of CSCW '96*.
41. Wolf, C., Rhyne, J. and Briggs, L. (1992) **Communication and Information Retrieval with a Pen-Based Meeting Support Tool Field Studies in Coordination**. *Proceedings of ACM CSCW'92 Conference on Computer-Supported Cooperative Work,* p.322-329.