Display and Presence Disparity in Mixed Presence Groupware

Anthony Tang, Michael Boyle, Saul Greenberg
University of Calgary
Calgary, Alberta, Canada
{tonyt, boylem, saul}@cpsc.ucalgary.ca

ABSTRACT

Mixed Presence Groupware (MPG) supports both co-located and distributed participants working over a shared visual workspace. It does this by connecting multiple single-display groupware workspaces together through a shared data structure. Our implementation and observations of MPG systems exposes two problems. The first is *display disparity,* where connecting heterogeneous tabletop and vertical displays introduces issues in how one seats people around the virtual table and how one orients work artifacts. The second is *presence disparity,* where a participant's perception of the presence of others is markedly different depending on whether a collaborator is co-located or remote. This is likely caused by inadequate consequential communication between remote participants, which in turn disrupts group collaborative and communication dynamics. To mitigate display and presence disparity problems, we determine virtual seating positions and replace conventional telepointers with *digital arm shadows* that extend from a person's side of the table to their pointer location.

Keywords: Mixed presence groupware, single display groupware, distributed groupware.

INTRODUCTION

The time/space taxonomy of groupware (Figure 1) categorises applications based on where and when collaborators use them (Baecker, Grudin, Buxton & Greenberg 1995). This introduces four quadrants defining styles of both groupware systems and work practices:

- same time / same place systems supporting face to face interactions,
- same time / different place systems supporting real time distributed interactions,
- different time / different place systems supporting asynchronous distributed work, and
- different time / same place systems supporting co-located on-going tasks.

Many applications have been designed to fit within a quadrant. MMM, for example, cleanly fits within the same time / same place cell because it supports co-located people sharing a single display using multiple mice (Bier & Freeman 1991). However, this quadrant view of groupware is limiting (Baecker 1993); in practice, people's collaborative practices cross these boundaries. For example, the rooms metaphor in TeamWave Workplace recognizes that people's collaboration with others may span the time boundary (Greenberg & Roseman 2003). Consequently, as multiple people enter a virtual room, they can interact synchronously over all items within a room. However, one can also leave items in a room for absent people to work on later, thus permitting asynchronous interaction.

	Same place	**Different place**
Same time	face-to-face interactions	real-time distributed interactions
	Mixed presence groupware	
Different time	co-located on-going work	asynchronous distributed work

Figure 1. Mixed presence groupware in the space/time groupware matrix

In the same vein, *mixed presence groupware* (MPG) supports both co-located and distributed participants working over a shared visual workspace in real time i.e., it spans the same place / different place quadrants at the top of Figure 1. Thus MPG defines synchronous groupware that is both distributed and co-located. Figure 2 gives an example, where the photos show several distributed groups of co-located people working over various physical displays containing a common shared visual workspace. As seen in the figure, the physical display may be a horizontal table-top display, or a vertical large presentation display (e.g., a projected display), or even a conventional monitor. All participants have their own input devices, and all can interact at the same time. Actions by participants are reflected on all displays. Conceptually, the physical tables embody a virtual table surrounded by co-present and remote participants (Figure 2, bottom right).

Our own interests are in the human, social and technical factors that arise in the design and use of these MPG applications by co-located and remote collaborators. In particular, our early implementations and observations of how people use our MPG prototype raised two problems.

1. ***Display disparity.*** Connecting heterogeneous tabletop and vertical displays introduces issues in how one seats people around the virtual table and orients work artifacts

Cite as: Tang, A., Boyle, M., & Greenberg, S. (2003). Display and presence disparity in mixed presence groupware. Report 2003-725-28, Department of Computer Science, University of Calgary, Calgary, Alberta Canada. August.

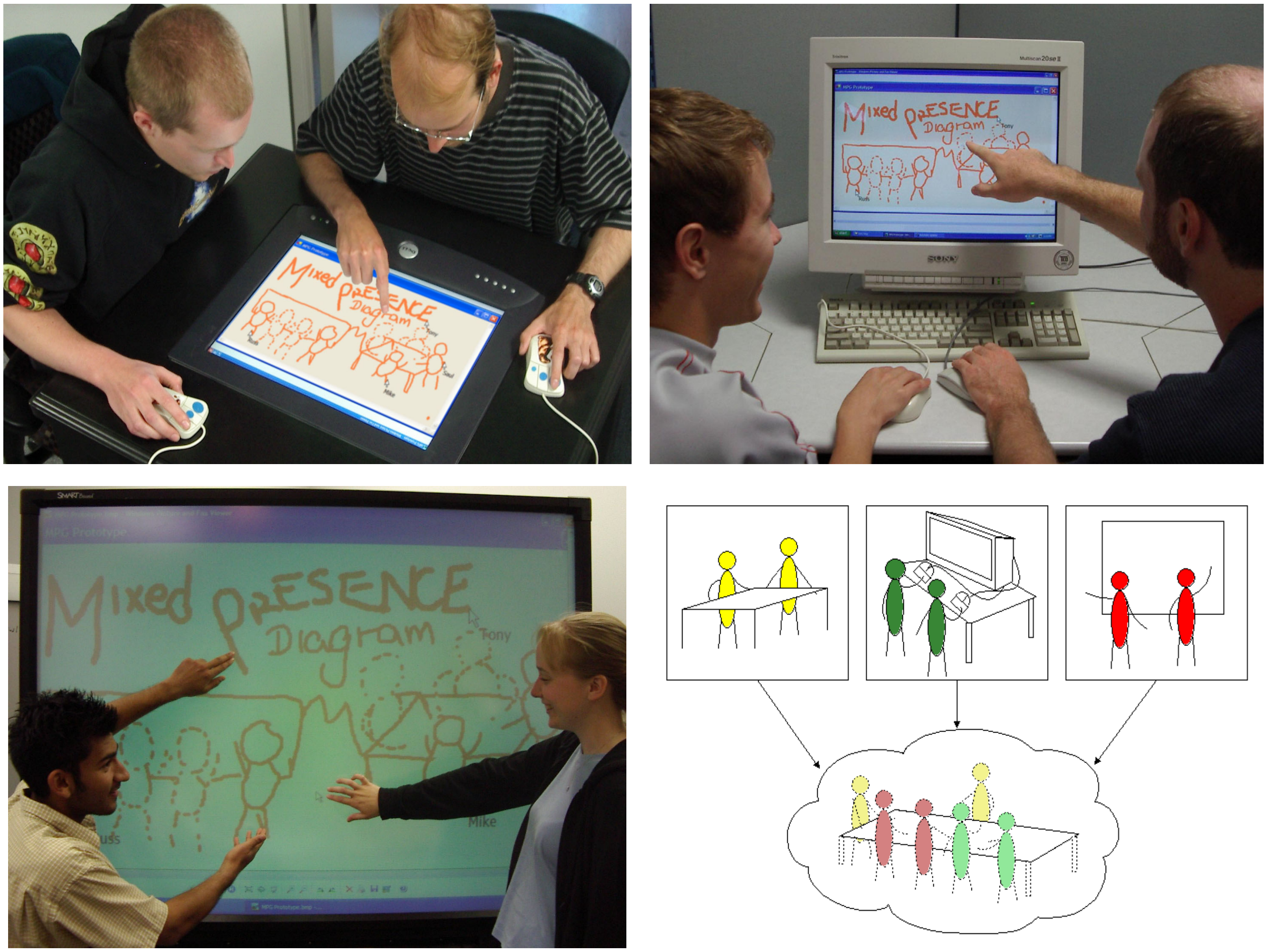

Figure 2. Three teams working in an MPG setting over three connected displays, stylized as a virtual table in the bottom right.

appropriately. For example, consider participants 1 and 2 working opposite each other on a table display, and a connected participant 3 working behind a monitor. The virtual table could seat all participants on separate sides, or have the participant 3 seated on the same side as participant 1. In either case, items drawn by participant 2 in his orientation will not appear 'right-side up' for participant 3.

2. ***Presence disparity.*** A participant's perception of the presence of others is markedly different depending on whether a collaborator is co-located or remote. This in turn disrupts group collaborative and communication dynamics. We suggest that one of its causes is that consequential communication (i.e., visibility of another's body) between remote participants is inadequate.

In this article we discuss our initial experiences in designing and building a mixed presence shared workspace groupware applications, and how we mitigate the display and presence disparity problem. We begin by situating mixed presence groupware within current groupware efforts. We next describe the iterative design and implementation of our prototype MPG application. We then discuss the human and technical aspects of presence and display disparity, garnered from our observations of the MPG prototype in use and from our technical experiences building these systems. Finally, we discuss techniques for linking heterogeneous displays, and introduce digital arm shadows as a method to restore presence parity.

RELATED WORK ON SHARED VISUAL WORKSPACES

A shared visual workspace is one where participants can create, see, share and manipulate artifacts within a bounded space. Real world examples are whiteboards and tabletops. Electronic counterparts to shared workspaces have been developed as distributed groupware, single display groupware, and to a much lesser extent mixed presence groupware.

Distributed groupware. Distributed groupware for shared visual displays abound, and has been a main focus for CSCW research over the past twenty years. These make interactions between distance-separated collaborators possible, and are attractive because they potentially reduce travel time and costs associated with remote collaboration.

For example, globally-minded enterprises are trying to use distributed groupware tools to assemble agile, cohesive and productive teams out of workers located in different cities and countries (Rogers 1994). Yet the design of these tools is fraught with social and technical challenges whose solutions are non-obvious. A large body of theoretical and empirical knowledge about these challenges has emerged from CSCW research into distributed groupware (Baecker 1993, Gutwin & Greenberg 2002) and several toolkits are now available to assist the researcher in rapidly prototyping distributed workspaces (Greenberg & Roseman 1999).

Single display groupware. While distributed interaction is clearly important, the bulk of a person's day-to-day interactions are co-located. This led to research into computer support for co-located interactions. In particular, single-display groupware (SDG) challenges the conventional 1:1 ratio between users and computers by allowing multiple users, each with his/her own input device (e.g., a mouse), to interact over a shared display (Stewart, Bederson, & Druin 1999). Early experiences with SDG systems indicate that they support natural dynamics of collaboration and conversation better than distributed groupware. Yet designing usable SDG interfaces and interactions is difficult. For example, hard technical factors include getting multiple devices to appear as independent input streams (Tse & Greenberg 2002). Hard social factors include recognizing and supporting the roles of orientation and personal space in mediating activity (Kruger, Carpendale, Scott, & Greenberg 2003). Although many important factors have yet to be thoroughly investigated, research into SDG has advanced to the point where there are now toolkits available to help rapidly prototype these kinds of systems (e.g. Tse & Greenberg 2002).

Mixed presence groupware. Given this research on both distributed and single-display groupware, one would expect equivalent advances in groupware that merges these concepts into MPG. Surprisingly, very few examples of this type of groupware exist in the literature. One is the Touch Desktop, created as part of the Swedish Institute of Computer Science's investigation into natural interaction within multi-user CAVE-like environments (Hansson, Wallberg & Simsarian 1997). As pictured in Figure 3, co-located people work on a touch screen tabletop display, which is placed in front of a 'communications wall' containing a 3d virtual environment. Actions on the physical table are reflected on the graphical table located in the virtual environment, and consequently visitors to the virtual environment can see what the collocated people are doing. However, the authors provide little additional information, and we suspect the system does not incorporate multiple physical tables.

A commercial example of MPG is Halo, a multi-player game for Microsoft's Xbox. Co-located players can interact through a split-screen, and distributed groups of players can be connected together by connecting several Xboxes together. All players and their actions are visible in each person's scene.

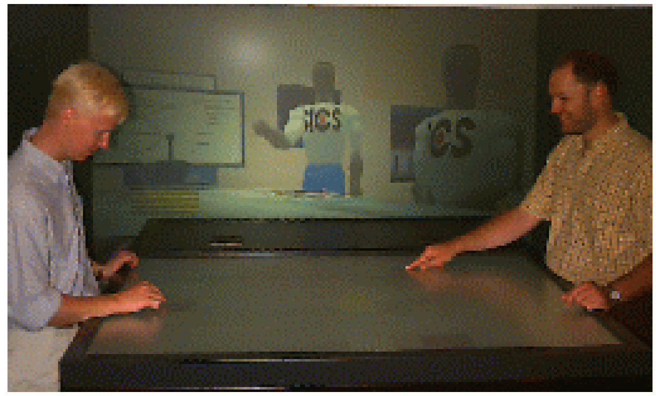

Figure 3. Touch Desktop. *Photo from Swedish Institute of Computer Science, www.sics.se/~par/dive_docs/interaction.html*

Perhaps the most common examples of MPG are based on video conferencing technology. A video channel captures and transmits co-located participants working over a drawing surface, or a special audio-graphics capability lets people annotate atop a video image. Some research systems even give people a shareable video-based drawing area by overlaying the images of two video cameras (e.g., Tang & Minneman 1991a+b; Ishii & Kobayashi 1992). While demonstrations typically show these as a means for connecting distributed people, co-located participants can be included simply by having them move into the scene. The catch is that the constraints of video overlays means that people cannot alter any artifacts on the drawing surface created by remote participants.

Finally, we should mention that people often work in an MPG mode even though their software may not support it. As a simple example, instant messengers explicitly support only one user per terminal chatting to others on their own terminals. However, others may chat "over the shoulder", by telling the co-located partner what to type, or by taking control of the mouse and keyboard.

Our focus on MPG is distinct from this prior work. First, we are interested in supporting how multiple co-located teams gain equal access to a single shared drawing surface. Second, all participants have their own input device, where each can manipulate the shared space—even simultaneously—at any time.

MPGSKETCH: A MIXED PRESENCE DRAWING SYSTEM

Our first goal was to understand the technical challenges of building MPG applications, and to gain some initial experiences in using one.

Description

We began our investigations by implementing and using MPGSketch, a simple MPG real time shared drawing application that collected distance-separated groups of co-located collaborators. Participants sketch over an empty surface, over an image taken from a file, or a video

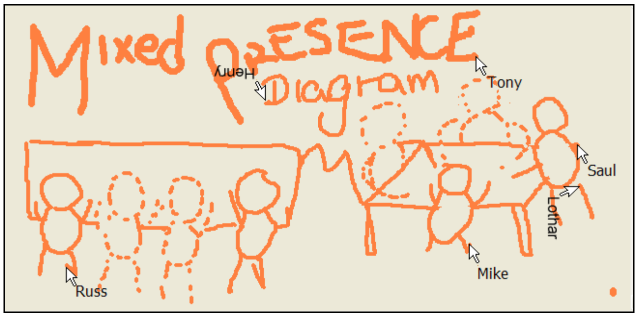

Figure 4. MPGSketch with six participants, each with a telepointer that reflects his or her local cursor position.

snapshot captured from a web-cam, or from a screen-grab of one person's desktop. A sample screen capture of MPGSketch is shown in Figure 4, and it is visible in action on the screens of participants in the Figure 2 photos.

Each person has his or her own pointing device for input e.g., a finger on a touch-sensitive table, a pen on a vertical whiteboard, or a mouse positioned near the front of the display. Each display presents the shared workspace containing the evolving drawing. Multiple cursors, labeled with their owner's name, show the location and movement of all pointing devices on this workspace. Any participant, whether local or remote, can draw on the display at any time, where their drawing actions can occur simultaneously. All drawing actions occur immediately on all displays. What makes MPGSketch an MPG application is that, as illustrated in Figure 2, several individuals can work on a single display, and that this display is connected to remote displays being worked on by other people.

Implementation

Because MPG applications are rare, it is worth taking a moment describing how we implemented MPGSketch.

We had two groupware toolkits at our disposal, both developed in our laboratory. First, SDGToolkit is a toolkit that makes it very easy to create single display groupware applications (Tse & Greenberg 2002). It recognizes multiple input devices (mice and keyboards) attached to a single computer, identifies their input events on a per-user level, and automatically gives feedback by drawing multiple cursors on the single display. It also manages tabletop applications, where a participant's mouse and cursor positions are automatically oriented towards their side of the table. However, SDGToolkit provides no support for distributed participants.

Second, the Collabrary is a toolkit that lets developers create multimedia groupware applications (Boyle & Greenberg 2002). At its heart is a well-developed API for capturing and manipulating multimedia data, and a means for easily sharing data between distributed processes through a shared dictionary. Developers typically create distributed groupware based on a distributed model-view-controller pattern. The shared dictionary is the common distributed model. Local inputs change the model, changes are propagated to all model instances as events, and local views are updated from the model.

We merged the capabilities of these toolkits to build MPGSketch. The SDGT toolkit takes care of managing the multiple keyboards/mice attached to a particular computer, and drawing the local cursors. It assigns each mouse a globally unique identifier and tracks the coordinates of its corresponding cursor. The MPGSketch instance then distributes this data via the Collabrary shared dictionary to other MPGSketch instances running on different computers. It stores mouse identifiers and it updates the cursors' on-screen coordinates as it moves. The remote MPGSketch instances (using the cursor component of the SDGToolkit) then draws cursors at the correct location for all of the remote input devices listed in the shared dictionary. Finally, as someone draws, the drawing coordinates are also placed in the shared dictionary. Based on this data, the MPGSketch instances update the drawing to give the shared view.

In principle, this implementation of mixed presence groupware is a reasonable approach for creating MPG applications. While our version depends on the SDGToolkit and the Collabrary, other tools with similar capabilities would suffice.

DISPLAY DISPARITY IN HETEROGENEOUS DISPLAYS

To help us understand MPG issues, we first tried to see what issues would arise if we ran MPGSketch across a heterogeneous display setting i.e., standard monitors, tabletops, and large displays. As we will see, connecting heterogeneous displays leads to *display disparity* that in turn introduces a number of issues.

- How does the system know where users are sitting around the horizontal display?
- How do we mechanically and visually orient pointing devices (e.g. mice) to reflect a participant's seating position? How should this orientation be treated on local *vs* remote displays?
- How do we manage 'non-upright' orientations on upright displays?
- How do we manage 'non-upright' orientations on remote horizontal displays?

The display disparity problems arise because, unlike monitors, tabletops have sides and lack an absolute notion of up and down. The notion of which side is 'up' is either undefined or arbitrary. Given this uncertainty, what does it mean to work around a table, and what does it mean to connect vertical monitors and horizontal tables?

Tabletop orientation. Unlike vertical displays, people can be seated across from one another or at right angles to each other around a table-top display. This introduces mechanical and visual orientation issues (Kruger, Carpendale, Scott & Greenberg 2003). Let us say that North is the traditional upright location. First, people in a non-North seat will be holding their mouse at a non-upright angle, which means that the coordinates returned when they mechanically move their mouse will be incorrect. Second, content (including labeled cursors) oriented correctly for one person will appear sideways or upside down to others. This problem is not particular to MPG—rather, it applies generally to table-top single-display groupware.

Fortunately, the SDGToolkit recognizes tabletop orientation. Each mouse can be associated with a side of the table (and implicitly, an orientation): North, South, East and West. All internal mouse coordinates are transformed relative to that orientation, so that the mouse behaves correctly for the user. Similarly, the labeled cursor is automatically oriented with respect to that orientation. However, it does not enforce any strategy for content orientation.

Heterogeneous orientation. While this strategy manages orientation within a single tabletop display, it does not solve the MPG-specific *display disparity* problems of what to do when multiple heterogeneous displays include both tabletop and vertical displays are connected.

What does it mean to connect vertical monitors with horizontal tabletops? One problem is that we need to establish their relative orientations. As a simplistic solution, we can assume that vertical monitors are always oriented to the North position, and arbitrarily assign a table a North position and demand that people work side by side at that position. However, this can result in 'overcrowding' of the North side (somewhat similar to Figure 2, bottom right).

Even if we do assign the North side to the vertical display, we are left with the problem of how to display other non-upright orientations. For example, South's cursors and actions will be upside-down, while East's and West's actions will be sideways (e.g., see Figure 4). While this is expected over tabletop displays, it looks decidedly odd—even unsettling—when this happens on a vertical display. We could translate cursors so they at least appeared right-side up on the vertical display but this would not work for items drawn on the surface that retain their orientation (e.g., text).

If we do not fix orientation, another problem is how people choose 'sides' of the virtual work surface. With joined tabletop displays, we need to at least determine which side is North. With vertical monitors, we need to specify what side of the virtual table corresponds to the bottom of the monitor. One strategy is to let people do this manually. Another strategy is to have the system assign sides e.g., to prevent overcrowding of any one side, it may try to balance people around the sides of the virtual work surface. Alternatively, it may try to favor a single side in order to give as many people as possible a common orientation.

EMBODIMENT AND PRESENCE DISPARITY IN MPG

Next, we conducted a very informal exploratory study of how two distributed groups used MPGSketch. To temporarily finesse the orientation issue, we used only upright monitors with a common 'North' orientation. We placed two pairs of participants (each knew the others well) in front of conventional workstation monitors on either side of a partition. Each workstation ran an instance of MPGSketch and had two attached mice. While people on one side of the partition could not see those on the other side, they could clearly hear them as they spoke. The four people then performed a non-competitive collaborative sketch task. While this experimental situation appears suspect—numbers are small and the task is uncontrolled—it was appropriate for our first foray into MPG use. We were looking for "big effects"—obvious issues, failures and successes—to guide our future investigations, and as typical in early testing, these are often seen in even very limited study situations.

All people were able to draw, and we saw no immediately obvious problems associated with the act of group drawing. This success is likely because we derived MPGSketch's design from a rich literature of observations of how people draw together (Tang 1991) and from our own experiences of similar systems supporting either remote or co-located drawing.

However, we were surprised to observe that most of participants' spoken utterances were directed towards their co-located partners. Rarely, if at all, did participants speak across the partition to the remote group. That is, there was a *conversational disparity* between co-located and remote participants. This is a major issue. To understand why conversational disparity occurred, we looked into the role of people's *embodiments* and the differences in *presence* it introduces in co-located / distributed real-time work.

Embodiments in the physical world

A person's body interacting with a physical workspace is a complex information source with many degrees of freedom. Bodily actions such as position, posture and movements of head, arms, hands, and eyes unintentionally "give off" information which is picked up by others (Baker, Greenberg and Gutwin 2001). This is a source of information, called *consequential communication,* for other co-located people since "watching other people work is a primary mechanism for gathering awareness information about what's going on, who is in the workspace, where they are, and what they are doing" (Gutwin 1997). Unintentional body language can be divided into three categories, as described below (Baker, Greenberg and Gutwin 2001).

Actions coupled with the workspace include gaze awareness (i.e. knowing where another person is looking), seeing a participant move towards an object or artifact, and hearing characteristic sounds as people go about their activities. This informs others of many things. First, one's body proximity to the workspace indicates whether they can see the contents of the workspace, their ability to actually reach into the workspace, and their orientation relative to the artifacts in the workspace. Second, body and hand motions tend to be large and take time to do, and this lets others infer and react to that person's intentions. For example, when others see a person's hand move over the drawing surface, they can anticipate what that person is about to do. They can then modify their own actions accordingly e.g., to avoid conflict, or to support the others' actions, or to repair potential problems before they occur.

Actions coupled to conversation are the subtle cues picked up from our conversational partners that help us continually adjust our verbal behaviour (e.g. Clark 1996). Some of these cues are visual ones coming from a person's embodiment: facial expressions, body language (e.g. head nods), eye contact, or gestures emphasizing talk. These visual cues provide *conversational awareness* that helps people nurture conversation. This in turn allows people to mediate turn-taking, focus attention, detect and repair conversational breakdown, and build a common ground of joint knowledge and activities (Clark 1996). For example, eye contact helps determine attention: people will start an utterance, wait until the listener begins to make eye contact, and then start the utterance over again (Goodwin 1981). On a coarser level, the proximity of a person's body to another person suggests different degrees of presence. This is important since presence is an essential cue used in initiating, continuing, and terminating conversation (Lombard & Ditton 1997). Many informal awareness cues for presence are visual in nature; for instance, people who are physically close are visually much larger than people who are far away. The visually large embodiments of co-located collaborators (compared to the telepointer embodiments of remote collaborators) make co-located collaborators appear comparatively more present.

While the above discussion deals with consequential communication, a person's embodiment also plays a significant role in *intentional communication*. These include explicit gestures and other visual actions used alongside verbal exchanges. For example, Tang (1991) observed that gestures play a prominent role in all work surface activity for design teams collaborating over paper on tabletops and whiteboards (around 35% of all actions). These are intentional gestures, where people used them to directly support the conversation and convey task information. Intentional gestural communication takes many forms (Baker, Greenberg and Gutwin 2001). *Illustration* occurs when speech is illustrated, acted out, or emphasized. For example, people often illustrate distances by showing a gap between their hands. *Emblems* occur when words are replaced by actions, such as a nod or shake of the head indicating 'yes' or 'no' (Short, Williams and Christie 1976). Deictic reference or deixis happens when people reference objects in the workspace with a combination of intentional gestures and communication, e.g., by pointing to an object and saying "this one" (Clark 1996).

Figure 5 brings these concepts to life. While we see only arms on the surface in this cropped photo, we immediately notice that two people are present, that both are poised to do work over specific places in different documents (by the position of the pen), and that the person on the left is pointing at an image with her pen and is emphasizing this with her other hand. The arm postures signal that both are engaged in this conversation.

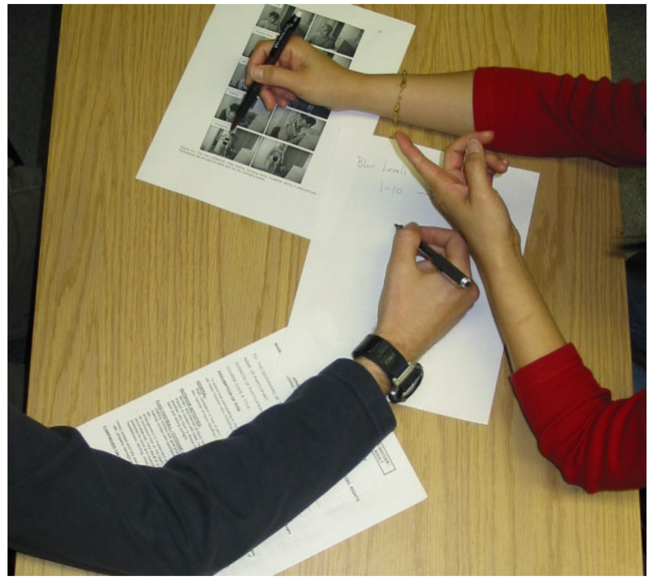

Figure 5. Corporeal arms in a common workspace.

Embodiments in MPGSketch

As with many real time groupware systems, MPGSketch provides all participants with multiple cursors (or telepointers). In distributed groupware, this small cursor (typically 32×32 pixels) is a remote user's only embodiment in the shared workspace when they are not actively drawing. While cursors are simple, they have proven effective in distributed settings. The presence and movement of the cursor serves as the visual representation of the distant person's presence and activity, and people are remarkably resilient at altering their work and conversational strategies to mitigate against the missing information.

The problem in mixed presence groupware is that there is a huge disparity between the embodiments of remote people (cursors), and the real-world embodiments of the local people (bodies). We call this difference *presence disparity.* For example, contrast people's real world arm embodiments in Figure 5 with the cursor embodiments in Figure 4. The size disparity alone is a major factor: arms are many orders of magnitude larger than a remote users' cursor, and thus commands much more attention. The low information richness and accuracy of the cursor embodiment is another disparity. For example:

- Cursors may suggest where its owner is looking but cannot guarantee it.
- An idle cursor (i.e., one that remains stationary for a while) suggests a person's presence, but again cannot guarantee it.
- The orientation of a cursor suggests where they are seated at a virtual table, but cannot indicate how the

person may actually be seated relative to that display in real life.

- Cursor gestures are reduced to deixis, with emblems and illustrations difficult to do.
- Cursors cannot transmit bodily proximity to others e.g., as happens in real life when a person leans in towards another to initiate conversation.
- While people normally initiate computer actions with their mouse, some cursor actions may be too quick or even invisible for others to see. This interferes with other's ability to infer intentions, and to react to them in a timely manner.

We believe that the presence disparity caused by the embodiment differences lead to the conversational disparity seen in mixed presence groupware. Because co-located embodiments dominate in presence through their size and richness, people direct nearly all of their utterances to co-located collaborators.

REBALANCING DISPLAY AND PRESENCE DISPARITY WITH DIGITAL ARM SHADOWS

We refocused our efforts in the second iteration of our MPG prototype to manage seating issues and to provide remote users with better embodiments.

Seating rules

Traditional groupware applications connect several upright displays together. The orientation of the shared workspace on these displays is identical: it would be odd to consider anything but a "North" orientation in these scenarios. In connecting upright and tabletop displays, display disparity means that some users at horizontal displays will invariably be at non-default (or non-North) orientations. Without special treatment, the model of the shared workspace and its participants would be as represented in Figure 2, lower right—a vast majority of users (those who are using upright displays) sitting at one side of the table with a given orientation, and a minority of users (a subset of those using horizontal displays) sitting at different sides of the table—each with a different orientation. While we do not know if this overcrowding is good or bad, we do believe that a few reasonable heuristics can help distributing participants around particular sides of a virtual table while preserving the physical orientation of co-located users.

1. Users' locations around physical tables are preserved around the virtual table.
2. Users who are seated side by side at an upright display remain seated next to one another at the virtual table.
3. Connected upright displays are automatically placed at different sides of the table.

Sensing User Presence

While we could let people choose sides through a dialog box, we instead designed two different implicit mechanisms to detect user presence. First, we recognize when a person sits on a particular chair around a table by embedding a light sensor in its seat and detecting when it goes dark (when one sits on it). We implemented this using Phidgets (Greenberg & Fitchett 2001). Of course, this solution requires fixed seating—since a seat is implicitly bound to some input device, moving seats around the table would require system recalibration.

Thus we developed a second implicit mechanism for detecting presence by monitoring mice movements, where each mouse is assigned to a particular seat. When people first sit down, they often wiggle their mouse rapidly to find their mouse pointer on-screen. We see this action as an informal way of greeting the computer—a presence signal. We detect absence through an inactivity timeout.

Of course, these two binary approaches to presence are somewhat simplistic as they are both prone to error. Also, a fairly large literature exists that conceives of presence as a deeper notion with many facets (for reviews, see Lombard & Ditton 2001) e.g., lurkers who watch but do not actively participate. However, we believe our approaches will work reasonably well in practice for most display scenarios.

With these methods of detecting presence in hand, we now discuss the digital arm shadows as the primary method for representing and presenting the presence information.

Digital Arm Shadows as Indicators of Social Presence

Once participants are seated, we now needed to communicate the orientation of each participant to others. For inspiration, we turned to VideoWhiteboard (Tang & Minneman 1991b), a video-based tool that provides a large shared drawing area between two sites. Video cameras behind the translucent drawing surfaces capture all activities on and near each surface, including not only the marks made on the surface with a felt pen, but a shadow of the body parts (usually hands and arms) as they move atop it. The video from both sites are then fused, creating a composite image. That is, the technology partially recreates the scene in Figure 4. Thus a person's arm gracefully appears as a shadow on the workspace as they move toward it and disappear as they move away from it. These arms were not only visually large; they were also socially natural indicators of presence. While extremely effective, VideoWhiteboard has technical limitations. It has high setup and equipment costs, people cannot edit each others marks, and it does not scale well because image degradation increases with the number of overlayed video streams.

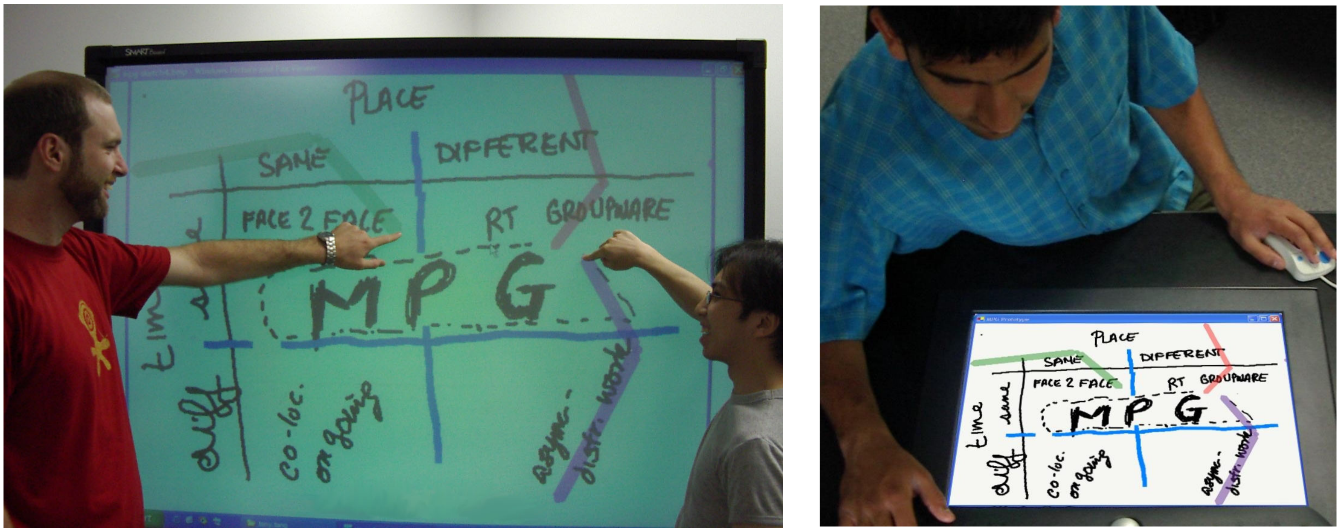

Figure 6. Presence with digital arm shadows

Although table-top and upright displays are not the same as whiteboards, we thought that arms might also make suitable embodiments in our MPG prototype. Consequently, we created *digital arm shadows* for remote collaborators that incorporated properties of presence seen in VideoWhiteboard. Using real arms working over a table as our model (such as Figure 5), each arm shadow maintains a 135° articulation and roughly maintain natural forearm/upper-arm, and width/length proportions. The "shoulder" point of an arm is attached to one of the sides of the table, and the "hand" point is bound to the mouse cursor location. The shadows themselves are semi-transparent, allowing objects on the underlying workspace to show through.

We packaged arm shadows as an independent software component (i.e., a 'widget') that we could incorporate into MPG applications. Through a simple programmatic interface, the programmer can bind the hand of the digital arms to telepointer locations, and the shoulder point to given positions around the display.

We then replaced MPGSketch's telepointers with arm shadows to represent participants. Figure 6 gives an example, with two people at the East and West side of a large display, and one person at the North side of a table. To show presence and absence, a shadow appears when a user's presence is detected, and disappears when one leaves. For example, when a person sits down at a chair, or begins using the mouse, the system conveys this presence information to all clients by drawing a corresponding arm shadow for that user. The system also conveys its uncertainty of one's actual presence by slowly increasing the transparency of the digital arm shadow when the owner is inactive. The software embodiment thus has a property of a real-life embodiment: the embodiment is only present when the person is physically present and active over the surface. In contrast to most other groupware systems, the system now differentiates between a person's presence at the terminal *vs.* a software client's connection to the system.

We then enhanced participant presence by creating a version of arm shadows that linked a live video portrait of each participant to their respective shoulders (Figure 7). For each participant, we captured a live video stream. Subtracting the background from this stream creates the small portraits, which we then orient and pin to the shoulder point of the appropriate arm shadow. This increases identity, and as allows other body language to come through the video. However, it does compromise space on the display.

To summarize, our contention is that arm shadows trigger the belief of remote collaborators' presence by reproducing

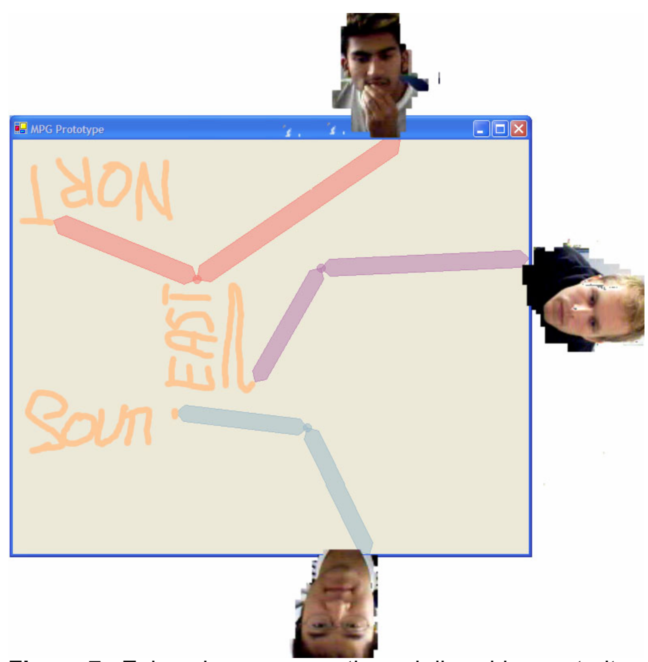

Figure 7. Enhancing presence through live video portraits

several key attributes of real-life embodiments (as in Figure 5) above and beyond those offered by standard telepointers.

- *Indicates virtual seating position*. Digital arm shadows appear from a side of the application window frame much as a person's corporeal arms are appear from a person's seating position. This "grounds" the virtual arms to an imagined virtual body.
- *Conveys person-specific orientation*. Each arm is has a different orientation, fostering the impression that each user has a distinct view of the display. This means that drawings oriented from that person are interpreted correctly (Kruger et al, 2003). For example, Tang (1991) noticed that drawings oriented towards its creator tend to be personal, while those oriented towards other tend to be public.
- *Increased awareness of actions.* A participant's actions are far more visible to others when compared to telepointers. First, our translucent shadows partially obscure the workspace underneath the arms, just as real arms obscure part of the table (Figure 5). Second, digital arm shadows are large (about an order of magnitude larger than telepointers).
- *Transmits identity.* People have extremely varied physical appearances—body/face size, shape and proportion, skin colour, hair, clothes, etc.—that are the essential cues for identity. Although our arms are far from photorealistic, they can be customized to approximate real arms and thus unambiguously represent other users. Current customizable arm parameters include color and proportion. Adding video portraits (Figure 7) increases identity substantially, at the cost of screen space.

These properties of the digital arm shadows, taken together, are virtualizations of real-life properties found in corporeal arms above and beyond those offered by standard telepointers.

DISCUSSION AND SUMMARY

The presence and display disparity problem we have discussed in this article is particular to mixed presence groupware systems. While the prototype MPG application presented is an example of an MPG shared visual workspace, acquiring and representing presence information appropriately is a general problem applicable to a wide array of distributed groupware systems. For example, signaling presence is an essential function of instant messaging systems (Nardi et al 2000). Also, collaborative virtual environments (e.g., Benford et al 1995) and media spaces (e.g., Gaver et al 1992) all seek to provide rich, socially natural embodiments for presence and informal awareness because, as suggested earlier, presence plays a vital role for regulating conversation. In tele-presentation and videoconferencing for distributed learning, local and remote audience members interact through video and audio links. Presence disparity in particular could negatively affect the learning experiences of students who must rely on the mediated link for interactions with their teachers. The TELEP system (Jancke et al 2000), for example, provided remote audience members with embodiments in a lecture theatre so that speakers could better field questions from remote viewers.

Our focus in this article was on dual co-located/distributed synchronous groupware, which we called mixed presence groupware (MPG). To help us understand design issues in this new class of groupware, we developed a prototype MPG groupware application which we hoped would afford users both the benefits of remote collaboration afforded by distributed groupware and the benefits of increased social interactions afforded by single-display groupware. Instead, we saw that most of our users' utterances are directed towards their co-located partners. We attributed this social dynamic to presence disparity: the presence of remote collaborators is weakly perceived relative to co-located collaborators. We believe that this diminished sense of presence impairs normal conversational dynamics. We also saw orientation problems arise from differences between display types and how we would seat people around the virtual table, which we called display disparity.

We adapted our prototype to work with mixed heterogeneous upright and table-top display configurations, where we handled participant seating and orientation. To the prototype we added digital arm shadows as a rich embodiment for presence. We chose digital arm shadows because they offered a variety of rich properties that we believe are important to signaling presence. We also added live video portraits of each participant to each arm. We believe that another person's physical presence triggers a set of mental processes that regulate social dynamics; our aim is to distill the numerous properties of physical presence to an essential subset required to trigger these mental processes—this false belief, of remote collaborators' presence.

Of course, these are early experiences in MPG. We have identified to critical factors—display and presence disparity—but there are likely other issues in MPG design. While we have demonstrated several 'solutions' to these issues, they are best considered design explorations rather than recommended practice.

ACKNOWLEDGEMENTS

This work was funded by the National Science and Engineering Research Council of Canada.

REFERENCES

1. Baker, K., Greenberg, S. and Gutwin, C. (2001) Heuristic Evaluation of Groupware Based on the Mechanics of Collaboration. In M.R. Little and L. Nigay (Eds) Engineering for Human-Computer Interaction (8th IFIP International Conference, EHCI 2001, Toronto, Canada, May), Lecture Notes in Computer Science Vol 2254, p123-139, Springer-Verlag.
2. Baecker R. (1992). Readings in Groupware and Computer Supported Cooperative Work. Morgan-Kaufmann, San Mateo, CA

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

4. Benford, S., Greenhalgh, C., Bowers, J., Snowdon, D., & Fahlén, L.E. (1995). User embodiment in collaborative virtual environments. In *Proceedings of CHI '95* (pp: 242-249). Denver: ACM Press.

5. Bier, E. A., & Freeman, S. (1991). MMM: A user interface architecture for shared editors on a single screen. In *Proceedings of the 4th annual UIST* (pp: 79-86). Hilton Head: ACM Press.

6. Boyle, M., & Greenberg, S. (2002). GroupLab Collabrary: A toolkit for multimedia groupware. In J. Patterson (Ed.). ACM CSCW 2002 Workshop on Network Services for Groupware, November.

7. Clark, H. (1996). Using Language. Cambridge University Press, Cambridge.

8. Gaver, W., Moran, T., MacLean, A., Lövstrand, L., Dourish, P., Carter, K., & Buxton, W. A. S. (1992). Realizing a video environment: EuroPARC's RAVE system. In *Proceedings of CHI '92* (pp: 27-34). Monteray, CA: ACM Press.

9. Goodwin, C. (1981). *Conversational Organization: Interaction Between Speakers and Hearers*, Academic Press.

10. Greenberg, S., & Fitchett, C. (2001). Phidgets: Easy development of physical interfaces through physical widgets. In *Proceedings of UIST 2001* (pp: 209-218). Orlando: ACM Press.

11. Greenberg S. and Roseman, M. (2003). Using a Room Metaphor to Ease Transitions in Groupware. In M. Ackerman, V. Pipek, V. Wulf (Eds) Sharing Expertise: Beyond Knowledge Management 203-256, January, Cambridge, MA, MIT Press.

12. Greenberg, S., & Roseman, M. (1999). Groupware toolkits for synchronous work. In M. Beaudouin-Lafon (Ed.), *Computer-Supported Cooperative Work (Trends in Software 7)* (pp. 135-168). New York: Wiley.

13. Gutwin, C. (1997). Workspace Awareness in Real-Time Distributed Groupware. Ph.D Thesis, Dept of Computer Science, University of Calgary, Canada.

14. Gutwin, C., and Greenberg, S. (2002) A Descriptive Framework of Workspace Awareness for Real-Time Groupware. Computer Supported Cooperative Work, 11(3-4), 411-446, Special Issue on Awareness in CSCW, Kluwer Academic Press.

15. Hansson, P., Wallberg, A., & Simsarian, K. (1997). Techniques for "natural" interaction in multi-user CAVE-like environments. Poster in ECSCW '97.

16. Ishii, H., & Kobayashi, M. (1992). ClearBoard: A seamless medium for shared drawing and conversation with eye contact. In *Proceedings of CHI '92* (pp: 525-532). Monterey: ACM Press.

17. Jancke, G., Grudin, J., & Gupta, A. (2000). Presenting to local and remote audiences: Design and use of the TELEP system. In *Proceedings of CHI 2000* (pp: 384-391). The Hague: ACM Press.

18. Kruger, R., Carpendale, M. S. T., Scott, S. D., & Greenberg, S. (in submission). How people use orientation on tables: Comprehension, coordination and communication. Used with permission of author.

19. Lombard, M., & Ditton, T. (1997). At the heart of it all: The concept of presence. *Journal of Computer Mediated Communication, 3*(2), from http://www.ascusc.org/jcmc/vol3/issue2/lombard.html.

20. Nardi, B. A., Whittaker, S., & Bradner, E. (2000). Interaction and outeraction: Instant messaging in action. In *Proceedings of CSCW 2000* (pp: 79-88). Philadelphia: ACM Press.

21. Rogers, Y. (1994). Exploring obstacles: Integrating CSCW in evolving organizations. In *Proceedings of CSCW '94* (pp: 67-77). Chapel Hill: ACM Press.

22. Short, J., Williams, E., and Christie, B. (1976). Communication Modes and Task Performance, in Readings in Groupware and Computer Supported Cooperative Work, R. M. Baecker Eds., Morgan-Kaufman Publishers, Moutain View, CA, 169-176.

23. Stewart, J., Bederson, B. B., & Druin,A. (1999). Single display groupware: A model for co-present collaboration. In *Proceedings of CHI '99* (pp: 286-293). Pittsburgh: ACM Press.

24. Tang, J. (1991). Findings from Observational Studies of Collaborative Work. *Intl J Man-Machine Studies*, 34(2), 143-160.

25. Tang, J. & Minneman, S. (1991a). VideoDraw: A Video Interface for Collaborative Drawing. *ACM TOIS Transactions on Information Systems,* 9 (2), April 1991

26. Tang, J. & Minneman, S. (1991b). VideoWhiteboard: Video shadows to support remote collaboration. In *Proceedings of CHI '91* (pp: 315-32). New Orleans: ACM Press.

27. Tse, E., & Greenberg, S. (2002). SDGToolkit: A toolkit for rapidly prototyping single display groupware. In *Extended Abstracts of CSCW 2002* (pp: 173-174). New Orleans: ACM Press.