Hootenanny – A Shared Workspace

Maria Freitas, Mark Leung, Michael Pennisi, Ricardo Sol
Vassilis Kostakos, Lucia Terrenghi, Pedro Campos
Carnegie Mellon University and Universidade da Madeira

This report proposes a shared remote application based on current practices in collocated collaboration. We design and implement features to enhance workspace awareness like temporary drawing and an overview. We perform heuristic evaluation, rapid iterative testing and evaluation, and a group experiment. These raise design implications for future collaboration studies.

Introduction

Our initial focus was broadly defined as follows: "Design a next-generation collaborative environment. This will consist of developing software for fixed and mobile devices to be used in a meeting room, as well as on fixed and mobile devices that are remotely connected." This statement required much refinement in order to serve as a guide for meaningful work; our refinement process is outlined below.

User Research

To understand current practices, we began by studying collaboration in real-world contexts. We purposely selected a diverse range of participants in order to identify practices in collaboration (and not practices of a specific target audience). This included participants from both the business world (i.e. lawyers, city planners, software developers) and from academia (i.e. students, professors, and education boards).

The methodology we employed to study each participant varied according to the nature of the work and their level of availability. Contextual inquiries (Beyer & Holtzblatt, 1997) helped us build a deep understanding of the work practices of eight local collaborators. We spoke with professionals located across Europe for whom collaboration was an essential component of everyday work. In these cases, we conducted telephone-based interviews to gain insights from the participants. A focus group consisting of fourteen professionals in human computer interaction gave us valuable perspectives on collaboration in today's business world.

Current Practices

Although we conducted our research with a broad focus, certain themes became apparent over time.

Our experience with the concept of "turn taking" in the domain of linguistic analysis sensitized us to parallels in collaborative work. We defined two "modes" of knowledge work: "turn-taking work" and "managed work". These delineated the collaboration practices we witnessed in terms of the coupling between colleagues. Managed work took place across independent workspaces, and a clear leader structured the efforts of all participants. Turn taking work occurred within a shared workspace with control shifting dynamically between all participants.

The practice of workspace subdivision was prevalent in our research. Collaborators sharing the same workspace would naturally section off areas for individual use. The motivation for this behavior varied. Some users desired a "private" space to complete an idea before displaying it. Some required a singular physical location on which multiple participants could focus. Some sectioned off the workspace to tackle sub-problems within the overall workspace. Nevertheless, we saw this basic practice repeatedly.

We also came to appreciate the complex (and largely implicit) practice of workspace awareness. It is the conceptual coherence of a shared workspace that differentiates it from multiple individual workspaces. This property necessitates that participants maintain an understanding of the space's constant flux. We noted two practices which achieved this goal: participants were vigilant to actions taking place in their periphery, and they repeatedly "stepped back" from their focus to view the workspace as a whole.

Shortcomings of Modern Solutions

In order to gauge how our perspective could inform novel solutions, we explored the state of modern-day shared workspace applications. We chose the “bodystorming” methodology (Simsarian, 2003) to motivate our exploration. In each application, we staged a critique of different web site designs. One team member presented an image of a web site design he or she was working on, and the rest of the team would review it and provide feedback. This method allowed us to experience multiple systems’ problems firsthand.

Frequently, the workspace grew beyond the size of our screens. Most solutions made it clear that more content was available off-screen (i.e. through the use of scroll bars), but they didn’t clearly express the distinction between populated areas and empty areas.

In order to efficiently move about the space, we needed an accurate model of our location relative to each other, the artifacts users had created, and the boundaries. Without such a model, the space felt somewhat nebulous. This, in turn, frustrated the processes of navigation and of communicating direction.

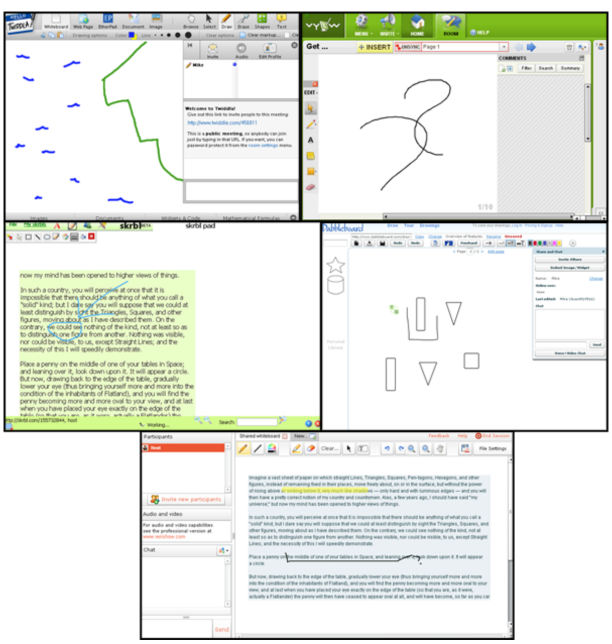

Figure 1. Some of the systems we investigated. Clockwise from the top-left: Twiddla, Vyew, Dabbleboard, ShowDocument, Skrbl.

Finally, every shared workspace represented a highly dynamic environment. Any user could make changes anywhere and at any time. In order to contribute in a meaningful and productive way, it was extremely important to maintain an understanding of what was happening at all times. Most systems ignored this necessity, leaving it to the users to coordinate their actions through other means (i.e. text chatting or voice).

Addressing the Shortcomings

Based on our experience using modern-day software, we drafted high-level features that would provide more assistance to collaborators.

Our user research and the relevant literature highlighted the importance of gaze in collaborative meetings. While gaze could be supported in this context with the use of eye tracking software, such an approach would also necessitate special browser plug-ins, and significant technological investment. Instead, we monitored the area each user’s browser was displaying. This information is readily available in all modern browsers and serves as a good approximation of a user’s focus.

We aimed to support the practice of workspace subdivision by recognizing clusters of users. By leveraging the location information required to approximate gaze, we could recognize when users were viewing the same areas, and group their representations in the workspace.

Pointing was another common task in real-world collaboration that required an analogue in electronic environments. Pointing is distinct from gaze because while gaze data represents an implicit cue towards a general area, pointing is an explicit indication of a specific location. In many systems we explored, pointing could be achieved by appropriating drawing functionality (i.e. drawing a circle around a misspelled word). We sought to support pointing’s unique properties that distinguished it from drawing.

2 Prior Work

2.1 Collocation Collaboration for Design Tasks

There has been a lot of work concerning the nature of collaboration in collocated environments.

Olson et al. (1992) studied the role of small meetings in supporting collocated collaboration in software design. Based on coded data from recordings of meetings, they found certain commonalities across diverse meetings. Teams devoted much of their time to managing their collaboration such as clarifying ideas or summarizing work; this would be difficult to do remotely.

Teasley et al. (2000) studied the benefits of "war rooms" – putting a project team in a single room for project duration – in aiding collaboration in software development. She found an increased ability to overhear each other that led to more spontaneous and frequent informal discussions. Large publicly-visible workspaces helped people coordinate their shared knowledge.

Wang and Blevis (2004) observed industrial designers' collocated collaboration around a common table. The table functioned as a shared workspace, where artifacts were visible accessible to everyone, and people could coordinate their activities through eye contact.

Tang et al. (2009) studied whiteboard use for supporting collocated collaboration, and found that the whiteboard's size afforded transitions between collaborative and independent work. The whiteboard's size let users all see the same workspace or have their own section to work on.

2.2 The Challenges of Distributing Collaboration

At the same time, researchers have weighed the potential of distributed collaboration with the challenges that come with removing physical copresence.

Olson and Olson (2000) studied a decade's worth of studies on distributed work. They noted both successes (space physicists collaborating remotely on data streams) and failures (software design). People would partition work to minimise remote interactions or change their behavior to compensate for poor communication channels. They concluded that successful distributed work depended heavily on establishing common ground (e.g. referencing the same artifacts and simultaneously receiving the same information). Furthermore, loosely coupled work that had few dependencies and was highly predictable was far easier to distribute. Collaborative design work is a tightly coupled task that is therefore difficult to do remotely.

Kiesler and Cummings (2002) pointed out that although groups were able to adapt to distributed work, mediated communication was time consuming and reduces social cohesion, feedback and cues. Groups have turned to managed independent work to circumvent the problems with mediated communication, but reduced lack of coordination with the rest of the team and increased bureaucracy. Collaborative design is a highly "disjunctive" task – one where a team solves a problem interdependently – and is especially reliant on coordination and cohesion.

Hinds and Bailey (2003) described increased conflict in distributed collaborative teams due to poor cohesion. Factors like common ground or familiarity were absent in remote collaboration; this increased the number of conflicts, and made these conflicts worse.

2.3 Shared Visual Space and Awareness

Researchers have explored shared visual spaces as a possible solution. Kraut et al. (2002) described a shared visual space as "one where multiple people can see the same objects at roughly the same time" They emphasize the benefits of shared visual space for visually complex tasks that are not easily described verbally. Researchers have applied this concept to annotation (Ellis & Groth, 2004), pen-based gestures(Ou et al., 2003), and virtual paper (Ashdown & Robinson, 2003), among other things.

One key hurdle in developing true replacements for colocated collaboration is translating the awareness that people have of peers in their physical surroundings.

Dourish and Bellotti (1992) referred to "awareness of individual and group activities": understanding the activities of others provides a context for your own activity. After observing shared editor use, they advocated maintaining awareness through shared feedback and showing individual users' activities on the shared workspace itself, in order to minimize work for the users sharing their contributions or monitoring others'.

Endsley (1995a) conducted research on situational awareness, which she defined as the "perceptions of the elements in the environment within a volume of time and space" (Endsley, 1995b). By studying situational awareness in the context of combat fighter pilots, she found that situational awareness of shared elements represents an index of team coordination. She modelled team awareness as the overlap between each member's individual situational awareness. This overlap could be acquired through viewing shared displays or verbal communication.

Gutwin and Greenberg (1997) worked on workspace awareness, the "knowledge about others' interaction with a shared workspace". In their work they proposed "awareness widgets": small onscreen devices that provide information on others' activities. (Gutwin, Greenberg, & Roseman, 1996; Gutwin, Roseman, & Greenberg, 1996; Gutwin & Greenberg, 1998, 1999) Examples of these included miniature view, which provided information about the state of the entire workspace, and radar view, which added information about the location of colleagues on the workspace.

3 Design Features and Interactions

The workspace supports several features that are aimed at allowing multiple people to work on a large workspace, while maintaining awareness of each other's actions. A workspace is shown in figure 2.

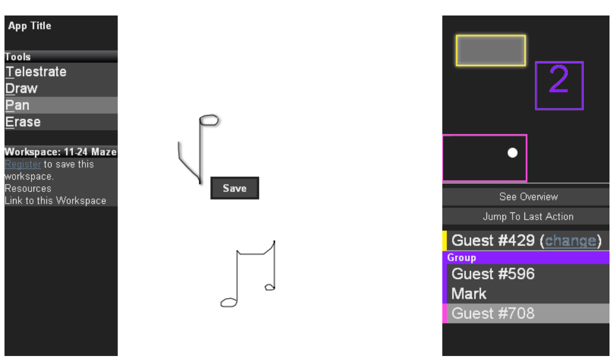

Figure 2. The prototype interface. Along with the workspace itself, note the the tool palette (left), minimap (right), temporary drawing (center), user list (right), and view boxes (within the minimap).

3.1 Tool Palette

The tool palette on the left of our workspace contains four tools which modify the behavior of the mouse. Users can select the draw tool to make marks on the workspace. They can change their view of the workspace using the pan tool. If they click the erase tool, they see erase buttons on each drawing they have made.

3.2 Temporary Drawings

Pointing and gesturing are important activities in face-to-face communication. People naturally use their hands to indicate the location of their subject and to describe certain details ("the fish was this big!").(Bekker et al., 1995)

These behaviors have analogues in digital workspaces, such as the drawing of simple shapes like circles, arrows, etc. While they accomplish a similar task, these "gestural" drawings differ from their physical analogs in being permanent. They outlive the discussion which they are tied to, only serving as clutter once the user has made his point.

It is from this prospective that we designed temporary drawings. The tool operates much like a "normal" drawing tool, but the lines it produce fade away once other participants make further temporary drawings. Users can optionally save a temporary drawing by designating it as permanent after creation.

Temporary drawings obviously have relevance to their intended audience, but because they do not reflect a permanent change to the workspace, they are not relevant to all participants. Therefore, the notification system does not alert all participants of new temporary drawings. This eliminates noise and can give the user more confidence in the relevance of each notification.

3.3 View Boxes

In face-to-face meetings, participants interpret each other's gazes to maintain coordination (Vertegaal, 1999). The "View Boxes" feature is an attempt to share this information remotely.

The area being viewed by each participant is laid atop the workspace. This helps the user to understand the focus of her colleagues. Some current solutions attempt to accomplish this task by exposing the location of all participants' cursors at all times (*Vyew*, n.d.). We find our approach to be less distracting (because viewing areas tend to be more static than cursor positions) and more expressive (because mouse positions do not accurately describe view boundaries).

View boxes require no explicit input from the user. By virtue of viewing the workspace, the participants provide this information to each other.

The system automatically recognizes when participants view the same section of the workspace. In these situations, proximate view boxes are displayed with a single rectangle approximating their overlap. An integer describing the number of members contained is also displayed. This representation calls attention to "groups" of users and clearly describes the number of participants. At the same time, any visual confusion resulting from similar overlapping view boxes is avoided.

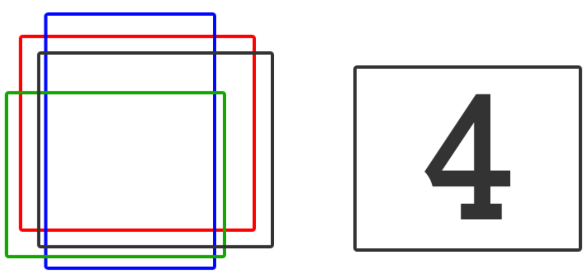

Figure 3. By distinctly visualizing users grouped in the same location, we can highlight their presence and member count while at the same time avoiding visual clutter.

3.4 Notifications

We consider a workspace to be an editable area defined by some conceptual focus. This focus means that any change to the workspace (no matter the location) may have implications for further changes. Therefore, when multiple people share the same workspace, it is important that they be aware of how it is changing over time.

In physical shared workspaces (i.e. meeting room white boards), participants can easily see where changes are happening, who is making those changes, and what those changes are. In digital settings, this information is not as easy to acquire. This is especially true when the workspace exceeds the viewable area on the participant's display. We provide this information in real time using notifications.

Each modification to the workspace triggers a notification in the user list. This notification is displayed alongside the user that made the change. As time passes, the notification fades from the user list. Rapid notifications generated by the same participant appear as one notification, to prevent information overload. The user may click on the user to move to the location of the change on the workspace.

3.5 Overview

The practice of "stepping back" from the workspace was prevalent in our research. Although people may prefer to focus on specific areas within their workspace, they frequently expand their focus in order to appreciate the entire area. Digitally supporting this action was a necessity.

The Overview function changes the relative scale that the workspace is drawn at. The entire workspace is thus "shrunken" in order to fit within the confines of the user's display. This gives the user a high-level description of the workspace as a whole. From this perspective, he may focus on any area within the workspace simply by selecting it with the mouse.

3.6 Minimap

Our ability to sense colleagues in the periphery aids work in shared physical space. The so-called "minimap" represents an effort to provide similar peripheral information within a digital space.

The minimap is a condensed representation of the entire workspace, fixed in the northeast corner of the interface. Because the contents of the workspace would be difficult to interpret at such a scale, they are omitted from the visualization. Only the viewing area of all active participants and groups appear in the minimap. This gives the user a general sense of her colleagues' location within the workspace.

4 Implementation

One can conceptualize the system is in three parts: the application running on the client's web browser, the routines executed by the server on behalf of the client, and the communication between these two parties. It is important to note that this relationship is identical for every participant in a multi-person workspace, and all the following functionality occurs concurrently whenever more than one user is present.

4.1 Client

The program running on the client's computer displays the workspace and captures the user's input. It is also responsible for detecting groups of users.

To display the workspace itself, the program uses the HTML5 "canvas" element as a surface onto which it "paints" graphical data (i.e. view boxes, user drawings, and event notifications). The surface is re-painted at a rate of twenty frames per second, simulating fluid animation.

There are a number of input methods available to the user. They may use the mouse to move around the workspace, to create new drawings, and to activate various buttons. They may also use keyboard keys as "shortcuts" for selecting workspace tools. All of this input is captured and interpreted using JavaScript.

When any user changes his position in the workspace, all participants immediately re-assess that user's group membership. "True" view-based groups (i.e. those defined solely by their participants' viewing areas) require maintaining overlap data—a computationally expensive task. In order to optimize the computation required of each participant, groups are approximated through the following heuristic.

The workspace is first subdivided into a grid of possible group locations. Next, each participant is assigned to a single grid location based on the center of his viewing area. Finally, users sharing the same grid location are joined into a group. This approach significantly reduces the computation required to simulate groups at the cost of some granularity (i.e. groups can only form in pre-determined locations).

4.2 Server

A dedicated web server acts as an arbiter for all participants. It receives each user's commands and maintains the workspace in a single consistent state. This task is accomplished with the PHP scripting language (Group, 2001)

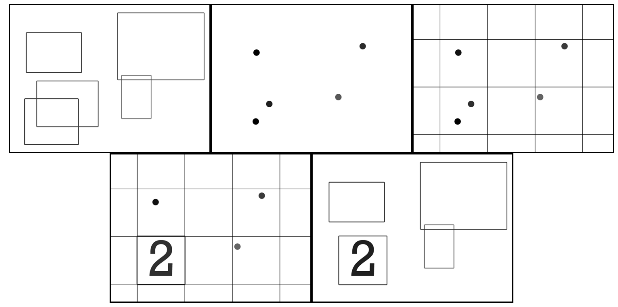

Figure 4. Clockwise from the upper-left: (1) The viewing areas of many participants (2) Simplifying participant representation (3) Subdividing the workspace (4) Grouping participants in the same section (5) The final visualization containing three single participants and one group

and the MySQL relational database management system (Corporation, n.d.).

The server uses a database to store a complete description of the workspace. This description includes details like the workspace dimensions, the drawings in the workspace, and the locations of the active participants. When a user first enters a workspace, the server sends all of this information so that the client program can accurately "paint" the workspace.

Whenever a user acts on the workspace (i.e. by moving their view or by creating a new drawing), the server receives the data describing this action. It uses this data to update the workspace. Finally, it informs every participant of this change, so that each client program can accurately re-paint its workspace.

The server leverages MySQL's "triggers" feature to efficiently broadcast changes to all participants. A trigger is a set of instructions that is automatically executed in response to some other database action. The server utilizes this behavior to automatically create "event" objects whenever a change is made to the workspace (such as the entry of a new participant or the deletion of a drawing). These events represent a unified interface for all changes to the workspace. This means that when a client asks for changes (through a process described in the following "Communication" section), it can simply ask for new events (as opposed to new drawings, modified drawings and deleted drawings).

4.3 Communication

Because this solution operates over the Internet, efficient communication between the client and server is essential. A user with a relatively low connection speed should be able to participate in the workspace. Inefficient bandwidth usage wastes resources for all parties, but the server is particularly sensitive because it maintains a number of separate communication links (one for each participant).

All data passed between the server and participants is formatted in JavaScript Object Notation (JSON). This format is human-readable and adds relatively little overhead to the data itself.

The technique AJAX (Garrett, 2005) is used to facilitate asynchronous communication between the client and server. This combination of technologies allows the client to continuously request new events from the server. In the current system, the client submits such a request once every 700 milliseconds. A high request rate guarantees that new events will always be promptly received by the participants.

Because users do not typically act on the workspace at this interval, the server responds with an empty message for most of these requests. These empty messages represent inefficiency in communication. A much more efficient solution would involve the participants passively waiting for messages sent by the server. Unfortunately, this technique is impossible with traditional web technology. "Web sockets" is an up-and-coming technology that can support such an approach.

Internet communication inherently suffers from a variable latency. While this latency is not perceptible for many interactions, it presented serious implications for our design of one feature: real-time access control. Because a great deal of work has already been done in this area (Greenberg & Marwood, 1994), and because it is only tangentially related to our focus, we elected to provide access control at the basic level necessary to maintain consistency between clients.

5 Heuristic Evaluation

Our first step of evaluation came while our system was still in the storyboard phase. Baker et al. (2001) adapted the concept of heuristic evaluation (Nielsen & Molich, 1990; Nielsen, 1992, 1994), and created a set of heuristics that was specific to groupware.

We used these heuristics to evaluate our design ideas and prototypes. Our goal was to look for holes in our solution in order to make it as complete as possible. Some of the heuristics were outside of the scope of our project and were therefore not considered. We found that our system satisfied the key applicable heuristics.

Gestural communication Our temporary drawing tool supports gestural communication. Like telepointer trails (Gutwin, 2001; Gutwin & Penner, 2002; Gutwin, 2002), the tool allows users to show their intentions over time. Unlike telepointers, however, temporary drawings let users specify when they want to communicate with others,

The individual's embodiment We intended to illustrate users' embodiment through their viewports (Gutwin, Roseman, & Greenberg, 1996; Gutwin, Greenberg, & Roseman, 1996). We extended this feature through the overview and the minimap, in which the whole workspace was visible and all viewports are visible at the same time.

Artifact Feedthrough Whenever a user makes a drawing, their name temporarily appears alongside that drawing. Their name in the user bar also highlights, giving other users a user-based notification of events.

The heuristic evaluation helped us identify certain missing component in our workspace as well. The heuristic "Management of Tightly and Loosely-Coupled Collaboration", led us to change our support for coupling. We had previously envisioned "following": one user would explicitly lock their view to another's when they wanted to collaborate on a section of the workspace, leading to a what-you-see-is-what-I-see effect. They would unlock their view once they were done.

The heuristic argued that having such a restrictive implementation would frustrate users, by poorly managing the transition between tightly-coupled (following) and loosely-coupled (independent) work. We instead designed a system that showed users as being in a group when they were viewing the same part of the workspace, without them explicitly locking their views to each other. This design managed coupling while reducing complexity of interactions for the user.

6 Iterative Testing

After producing a functional prototype, we used a rapid iterative testing (Medlock et al., 2002) approach to quickly refine our design.

Participants

We recruited 10 masters students at the University of Madeira. Our subjects ranged from 22 to 30 years of age. There were both native and second language English speakers in the subject group.

Procedure

We conducted some of our testing in groups to simulate our collaborative use cases. The participants were located in the same room, but were arranged so that they could not see each other's screens. We also ran studies with individual users, where we collaborative tasks alongside the participants. This enabled us to gain more insights and opinions more quickly at the expense of the experimenters guiding the user towards their style of use.

We employed several different tasks to motivate our user testing. We favored tasks that were open ended and required building on the work of others.

Mindmap Users were instructed to produce a brainstorming mind map around a topic provided by the experimenter.

Drawing Users were instructed to work as a group to complete a drawing of an object suggested by the experimenter.

Maze Users were instructed to begin a different places in a maze and try to negotiate a meeting place with each other.

Selection Users were instructed to view a set of objects on the shared workspace and choose the ones they liked as a group.

After conducting the experiment with a group of participants, we conducted a focus group where users could discuss their experience and make suggestions. We recorded the activity of individual users with commercial usability testing software and performed retrospective interviews. Some of our observations are described below.

Findings

Spatial Awareness of the Workspace. Users did not think of the workspace like they would a physical workspace. Their preferred method of sensing the dimensions of the workspace was to enter the overview mode and then reenter the workspace. When viewing the workspace at normal scale, they simply saw their view as the workspace and worked in that part. Once they had filled that space, they would move as a group to the next available patch of workspace. This was especially evident in the scale of participants' drawings during tasks like mindmapping. In these cases, users consistently drew at a scale relative to the size of their viewable area. For instance, users did not produce larger drawings when the workspace immediately surrounding them was empty. Conversely, they did not produce smaller drawings as their viewing area approached workspace boundaries.

Users tend towards loosely-coupled collaboration. We saw users naturally work very separately. Even though users could effectively make use of the awareness tools when prompted to, they normally chose to work more independently than had they performed the corresponding task in the physical world.

User-based Notification. Users had feedback about our notification system, explaining that "you want to pan not to the person, but the activity". During our design process, we had designed the system such that notifications were user centric. Since the user list played a large role in showing which people were working together, we assumed that using it as a notification source would simplify the user's mental model. Instead, participants found the user-oriented feedback confusing. They ignored the highlighting that occurred when a user was active since it failed to communicate the location or content of the event.

Mental reorganization of the radar view and mini map. Users had suggestions about our minimap. Whereas the overview provided an complete view of the workspace, the minimap showed a far more simplistic view of the workspace. All contributions were hidden, and users only saw other users' viewboxes. Their own viewbox was distinguished by a "glowing effect".

Users complained of the difficulty of translating the minimap to their location. They had to look between the user list, workspace and the minimap to orient themselves, much like the way that people have to translate a map's representation to their physical position when map reading. ()

Temporary drawings - gauging when to fade. Users had difficulties predicting the behaviour of temporary drawings. They found that the temporary drawings often faded before they would have liked. Sometimes they would fade before they could show another user what they had done, while others, . This was especially evident during periods of high latency in the system.

7 Quantitative Evaluation

We augmented our qualitative observations with a quantitative study measuring how much certain features in our solution helped improve group performance in a collaborative task.

7.1 Participants

20 people participated in our study. All the participants were students or researchers at the University of Madeira. They were aged from 20 to 30.

7.2 Conditions

We created two conditions. In the experimental condition, the full interface was available to the participants. In the control condition, the minimap and notifications were disabled in the interface.

7.3 Choice of Task

We used a picking task for our quantitative evaluation. Subjects were presented with a selection of pictures of cupcakes on the shared workspace. Each cupcake had a price value attached to it ranging from two to four euros. Subjects were instructed to evaluate the cupcakes as a group and select those that they preferred while staying within a budget of 20 euros. They were disallowed from choosing a subset of cupcakes e.g. green cupcakes.

This task simulated the process people go through during a design task, in having them discuss their opinions of shared artifacts in light of certain constraints. Better collaboration would lead to faster discussion times and closer adherence to the specified budget.

7.4 Stimuli

We created four images containing 25 randomly-distributed photographs of cupcakes. Each image consisted of 10 cupcakes that fit a certain constraint, and 15 cupcakes that did not. The four constraints were: green cupcakes, cupcakes with animals, cupcakes with brown icing, and cupcakes with flowers.

Figure 5. One of the images used as stimuli in our evaluation. Here, participants were restricted from selecting cupcakes with flowers.

Randomisation

We randomized the experiment procedure on two dimensions. One dimensions concerned which interface (control or experimental) subjects were asked to use first. The other dimension concerned the ordering of the four image stimuli.

7.6 Procedure

Subjects were tested in groups of four, in a collocated environment. Each subject was given a laptop computer connected to the workspace. They were seated such that they could not see each other's screens, although they could see each other. They were told that they were in a study testing their ability to come to a consensus and received a brief tutorial on the use of the software.

Each group of subjects completed four trials. Subjects were timed from when they entered the workspace until they informed the experimenter that they had completed the task. The recorded their choices and the total price of these choices.

After the second and fourth trials (finishing the control and experimental conditions), users were asked to fill out a questionnaire. This questionnaire contained ten subjective questions (presented on a Likert scale (Likert, 1932)) regarding the utility and usability of the interface.

7.7 Results

	Control	Experiment	Significance
Time	385.4	422.9	n.s.
Closeness	0.2	0.5	n.s.

Table 1

T-test of time to complete task and closeness to monetary goal

	Control	Experiment	Significance
Usefulness	4.4625	4.1	n.s.
Ease of use	3.991666667	5.05	n.s.

Table 2

T-test of questions assessing usefulness and ease of use

We performed t tests on the time taken to finish each trial, and on the closeness of the money spent on cupcakes to the goal of 20 euros. We found a difference between the control and experimental conditions, but it was not significant. We also performed a t test of the questionnaire questions. We separately evaluated questions related to usefulness and ease of use of our system. Neither difference was significant.

8 Design Implications

Throughout our evaluation, we witnessed a preference for loosely-coupled collaboration. The tendency for participants to favor such styles has been well-documented in the evaluation of groupware (Olson 2000). Despite this behavior, we have recognized promising future work in the development of the system.

8.1 Location-Based Notifications

Many participants supplied us with feedback relating to being at the conceptual "center" of the interface. While the exact wording varied between participants, common throughout was a strong desire to keep oriented within the workspace. Moving forward, we feel that location-based notifications may aid in this task.

Participants were more interested in where events took place than the colleague that they originated from. Because of this, we recommend positioning the notifications relative to the location of the event. We call this scheme "edge pointers", because each notification is displayed along the edge of the screen closest to the event.

In the small group environments in which we tested our solution, a location-based notification system seemed more appropriate than a user-based system. Because users could (and frequently did) work with all their colleagues at the same time, it was less necessary to tie notifications to the person who performed each action. Further, users were less likely to be overloaded by notifications because of their relatively slow rate.

Location-based notifications do not scale well for rapid notifications, i.e. large group workspaces. It is easy to imagine how a user could be "swamped" by a rapid succession of new notifications. In order to facilitate larger groups with location-based notifications, grouping mechanisms should be explored. For instance, instead of visualizing each notification separately, notifications originating from similar locations could be displayed together.

8.2 Temporary Drawings

In developing the prototype, we were interested in temporary drawings to the extent that they supported workspace awareness and coordination. Their novelty extends beyond this domain, and the way they were utilized during our user studies proved this. Users were generally unsure of how the temporary drawings were managed—how they were created, when they would be removed, how they related to other temporary drawings, etc. Unfortunately, before they can successfully enhance workspace awareness, the behavior of the temporary drawings must be refined.

We feel a dedicated study would be appropriate to design a truly useable temporary drawing feature. We see many possible implementation schemes, and finding the best scheme warrants a dedicated research effort. This research may take place outside the context of workspace awareness and instead focus on usability, feed-forward, and transparency studies.

This future investigation could have implications for shared drawing applications.

9 Conclusion

Our goal during this project was to create a complete working solution to enable a shared remote workspaces. In doing our research, competititve analysis, and literature, we were able to generate design insights that let us to propose design ideas and concepts. Given the limited time to iterate and the difficulty of evaluating low fidelity prototypes of a shared workspace, we made several key design assumptions and performed a design process based on them.

Overall, we successfully produced a working solution, one that did not show significant difference in performance from a control. Our qualitative evaluation showed some successes, while raising questions about temporary drawings, notification, and remote collaboration itself; they present interesting opportunities for future work. We also see possibilities for reusing and extending our tool to help test out existing and new features in more detail.

References

Ashdown, M., & Robinson, P. (2003). The escritoire: remote collaboration in a task space. In *Proceedings of the 2003 acm sigmm workshop on experiential telepresence* (pp. 73–75).

Baker, K., Greenberg, S., & Gutwin, C. (2001). Heuristic evaluation of groupware based on the mechanics of collaboration. In M. Little & L. Nigay (Eds.), *Engineering for human-computer interaction* (Vol. 2254, p. 123-139). Springer Berlin / Heidelberg. (10.1007/3-540-45348-$2_1$4)

Bekker, M. M., Olson, J. S., & Olson, G. M. (1995). Analysis of gestures in face-to-face design teams provides guidance for how to use groupware in design. In *Proceedings of the 1st conference on designing interactive systems: processes, practices, methods, & techniques* (pp. 157–166). New York, NY, USA: ACM.

Beyer, H., & Holtzblatt, K. (1997). *Contextual design: Defining customer-centered systems*. Morgan Kaufmann, San Francisco CA.

Corporation, O. (n.d.). *Mysql :: The world's most popular open source database.*

Dourish, P., & Bellotti, V. (1992). Awareness and coordination in shared workspaces. In *Proceedings of the 1992 acm conference on computer-supported cooperative work* (pp. 107–114). New York, NY, USA: ACM.

Ellis, S. E., & Groth, D. P. (2004). A collaborative annotation system for data visualization. In *Avi '04: Proceedings of the working conference on advanced visual interfaces* (pp. 411–414). New York, NY, USA: ACM.

Endsley, M. R. (1995a, March). Measurement of situation awareness in dynamic systems. *Human Factors: The Journal of the Human Factors and Ergonomics Society, Volume 37, Number 1, March 1995 , pp. 65-84(20)*, *37*(1), 65-84(20).

Endsley, M. R. (1995b, March). Toward a theory of situation awareness in dynamic systems. *Human Factors: The Journal of the Human Factors and Ergonomics Society, Volume 37, Number 1, March 1995 , pp. 32-64(33)*, *37*(1), 32-64(33).

Garrett, J. J. (2005). Ajax: A new approach to web applications.

Greenberg, S., & Marwood, D. (1994). Real time groupware as a distributed system: concurrency control and its effect on the interface. In *Proceedings of the 1994 acm conference on computer supported cooperative work* (pp. 207–217). New York, NY, USA: ACM.

Group, T. P. (2001). *Php: Hypertext preprocessor.*

Gutwin, C. (2001). Traces: Visualization of interaction. *Department of Computer Science, University of Saskatchewan, Saskatoon, SK, Canada.*

Gutwin, C. (2002). Traces: Visualizing the immediate past to support group interaction. In *Graphics interface* (pp. 43–50).

Gutwin, C., & Greenberg, S. (1997). *Workspace awareness in real-time distributed groupware*. Citeseer.

Gutwin, C., & Greenberg, S. (1998). Effects of awareness support on groupware usability. In *Proceedings of the sigchi conference on human factors in computing systems* (pp. 511–518).

Gutwin, C., & Greenberg, S. (1999). The effects of workspace awareness support on the usability of real-time distributed groupware. *ACM Transactions on Computer-Human Interaction (TOCHI)*, *6*(3), 243–281.

Gutwin, C., Greenberg, S., & Roseman, M. (1996). Workspace awareness support with radar views. In *Conference companion on human factors in computing systems: common ground* (p. 211).

Gutwin, C., & Penner, R. (2002). Improving interpretation of remote gestures with telepointer traces. In *Proceedings of the 2002 acm conference on computer supported cooperative work* (pp. 49–57).

Gutwin, C., Roseman, M., & Greenberg, S. (1996). A usability study of awareness widgets in a shared workspace groupware system. In *Proceedings of the 1996 acm conference on computer supported cooperative work* (pp. 258–267).

Hinds, P., & Bailey, D. (2003). Out of sight, out of sync: Understanding conflict in distributed teams. organization science, 14(6), 615-632. *Organization Science*, *14*(6), 615-632.

Kiesler, S., & Cummings, J. (2002). What do we know about proximity and distance in work groups? a legacy of research. In P. Hinds & S. Kiesler (Eds.), *Distributed work* (p. 57-82). Cambridge, MA: MIT Press.

Kraut, R. E., Gergle, D., & Fussell, S. R. (2002). The use of visual information in shared visual spaces: informing the development of virtual co-presence. In *Cscw '02: Proceedings of the 2002 acm conference on computer supported cooperative work* (pp. 31–40). New York, NY, USA: ACM.

Likert, R. (1932). A technique for the measurement of attitudes. *Archives of Psychology*, *22*(140), 1–55.

Medlock, M., Wixon, D., Terrano, M., Romero, R., & Fulton, B. (2002). Using the rite method to improve products: A definition and a case study. In *Proceedings of upa.*

Nielsen, J. (1992). Finding usability problems through heuristic evaluation. In *Proceedings of the sigchi conference on human factors in computing systems* (pp. 373–380). New York, NY, USA: ACM.

Nielsen, J. (1994). Usability inspection methods. In *Conference companion on human factors in computing systems* (pp. 413–414). New York, NY, USA: ACM.

Nielsen, J., & Molich, R. (1990). Heuristic evaluation of user interfaces. In *Proceedings of the sigchi conference on human factors in computing systems: Empowering people* (pp. 249–256). New York, NY, USA: ACM.

Olson, G. M., & Olson, J. S. (2000). Distance matters. *Human Computer Interaction*, *15*, 139-179.

Olson, G. M., Olson, J. S., Carter, M. R., & Storrøsten, M. (1992). Small group design meetings: an analysis of collaboration. *Proceedings of the 1992 ACM Conference on Human Factors in Computing Systems*, *7*, 347-374.

Ou, J., Chen, X., Fussell, S. R., & Yang, J. (2003). Dove: drawing over video environment. In *Multimedia '03: Proceedings of the eleventh acm international conference on multimedia* (pp. 100–101). New York, NY, USA: ACM.

Simsarian, K. T. (2003). Take it to the next stage: the roles of role playing in the design process. In *Chi '03 extended abstracts on human factors in computing systems* (pp. 1012–1013). New York, NY, USA: ACM.

Tang, A., Lanir, J., Greenberg, S., & Fels, S. (2009, May 10-13). Supporting transitions in work: Informing large display application design by understanding whiteboard use. In *Proceedings of the acm international conference on supporting group work - acm group'09* (p. 149-158). ACM Press. (Earlier different version available as Report TR-2008-04, Dept Comp Science, UBC (2008))

Teasley, S., Covi, L., Krishnan, M., & Olson, J. (2000). How does radical collocation help a team succeed? *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work*, 339-346.

Vertegaal, R. (1999). The gaze groupware system: mediating joint attention in multiparty communication and collaboration. In *Proceedings of the sigchi conference on human factors in computing systems: the chi is the limit.*

Vyew. (n.d.).

Wang, H., & Blevis, E. (2004). Concepts that support collocated collaborative work inspired by the specific context of industrial designers. *Proceedings of the 2004 ACM conference on Computer supported cooperative work*, 546 - 549.