

Socially Translucent Systems: Social Proxies, Persistent Conversation, and the Design of "Babble"

**Thomas Erickson, David N. Smith, Wendy A. Kellogg,
Mark Laff, John T. Richards, Erin Bradner***

IBM T.J. Watson Research Center

P.O. Box 704

Yorktown Heights, NY 10598, USA

+1 914 784-7826

snowfall@acm.org, dnsmith@watson.ibm.com, wkellogg@us.ibm.com,
mrl@us.ibm.com, jtr@watson.ibm.com, ebradner@ics.uci.edu

ABSTRACT

We take as our premise that it is possible and desirable to design systems that support social processes. We describe Loops, a project which takes this approach to supporting computer-mediated communication (CMC) through structural and interactive properties such as persistence and a minimalist graphical representation of users and their activities that we call a social proxy. We discuss a prototype called "Babble" that has been used by our group for over a year, and has been deployed to six other groups at the Watson labs for about two months. We describe usage experiences, lessons learned, and next steps.

Keywords

Conversation, Discourse, Awareness, Social Activity, Computer-Mediated Communication, CMC, IRC, Chat, CSCW, Social Computing, Design, Visualization

INTRODUCTION

In the building where our group works there is a door that opens from the stairwell into the hallway. This door has a small design flaw: opened quickly, it is likely to slam into anyone who is about to enter from the other direction. In an attempt to fix this problem, a small sign was placed on the door: it reads, "Please Open Slowly." As you might guess, people soon ceased noticing the sign and its effectiveness decreased markedly.

We would like to contrast this solution with one of a different sort: putting a glass window in the door. The glass window approach means that the sign is no longer required. As people approach the door they can see whether there is anyone on the other side. This is effective for three reasons: First, as humans, we are *perceptually attuned* to movement and human faces and figures, and notice and react to them more readily than we notice and interpret a printed sign. Second, the glass window supports a perceptually based *awareness*: I don't open the door quickly because *I know* that you're on the other side. This *awareness* brings our social rules into play to govern our actions: we have been raised in a culture which frowns upon slamming into other people (except in narrowly defined, culturally recognized situations). Finally, there is a third, and somewhat subtler reason for the efficacy of the glass window. Suppose that I don't care whether I harm others: nevertheless, I am still likely to open the door slowly because *I know that you know that I know* you're there, and therefore I will be held *accountable* for my actions. This distinction is useful because, while accountability and awareness are generally entwined in the physical world, they are not necessarily coupled in the digital realm.

We call systems which provide perceptually-based social cues which afford awareness and accountability "Socially Translucent Systems." In such systems we believe it will be easier for users to carry on coherent discussions; to observe and imitate others' actions; to engage in peer pressure; to create, notice, and conform to social conventions. We see social translucence as a fundamental requirement for supporting communication and collaboration. This brings us to the issue of translucence. Why Socially *Translucent* Systems? Because there is a vital tension between privacy and visibility. Neither is inherently good or bad: each supports and inhibits certain types of behavior (for example, the perceived validity of elections depends crucially on keeping certain aspects very private, and others very visible).

The basic premise of our work is that it is possible and desirable to build digitally based, socially translucent systems which allow our socially based processes to operate. In this paper we describe a system that takes this approach to supporting computer-mediated conversation.

Related Work

A concern with making the activity of users of digital systems visible to others dates back to at least the Finger program on UNIX. More recently, a number of investigators have explored ways of portraying socially-salient information in human computer interfaces. Ackerman and Starr [1] have argued for the importance of social activity indicators, particularly in synchronous CMC systems. Hill and Hollan have discussed the creation of persistent traces of human activity [14]. And many researchers have constructed systems which attempt, in various ways, to provide cues about the presence and activity of their users (for example, [3, 6, 15, 16, 18, 20]). Closest in spirit to our approach are the Out to Lunch system [5], and the AROMA [19] and Chat Circles [7] systems, which use abstract representations (sonic and visual, respectively) to portray social activity.

THE LOOPS PROJECT

Our goal is to design systems that support smooth, reflective, and productive conversations through synchronous and asynchronous computer-mediated communication. We believe that social translucence is particularly applicable to the design of CMC systems. After all, conversation is a fundamentally social process. Face to face conversation relies heavily on social cues: facial expressions, gestures, postures, and other socially-relevant actions are used in the initiation and conduct of conversation [12, 13, 17].

That is, we speak *to* an audience: nods and eye contact convey one message, yawns and fidgeting another, and watching people slip out of the room during a presentation is a powerful motivator to either change course or wrap up.

This raises the question of what sorts of social cues might be useful in supporting CMC, and how they might be presented. An obvious answer is to explore the use of video or high resolution virtual reality to depict the subtleties of facial, gestural and postural expressions. We have rejected this approach for two reasons. First, we believe that systems which attempt to leverage social processes need to be developed through a process of creating and deploying working systems, and studying their use in ordinary work contexts. This intent to deploy, in and of itself, argues against a 'radical technology' approach such as VR. Second, using video (or pictures) of participants' faces takes up significant screen space, and doesn't scale to larger groups. We also find the prospect of digital environments populated by floating heads to be aesthetically unappealing. So, instead, we decided to take a minimalist approach to providing socially salient cues.

"Loops" is the project name and serves as a catch all for the assumptions, design rationales, and conceptual and working prototypes developed in this project. Loops is not an acronym, but is taken from the idiomatic phrase, "keep me in the loop," which refers to keeping participants aware of what is going on. The premise of Loops is that conversation is a powerful tool for creating, developing, and sharing knowledge. The ability to carry out coherent, productive conversations among many participants distributed in time and space is likely to become an increasingly important organizational skill. The goal of Loops is to make it easy and practical to initiate, conduct, and share such conversations, and the knowledge developed through them.

A Brief Project History

Although we will focus on the working prototype and its long-term usage, it is useful to situate it in a larger context. Of particular import is the fact that Loops was developed and used within a closely knit work group (aka "the lab"). The work group had two remote members and a number of other members who were 'locally mobile' [2], i.e. often not at their desks.

The Loops concept was initially developed using storytelling, scenarios, and rough prototypes [8]. Beginning with stories about mailing list use (e.g., floods of "Please unsubscribe me messages" as cues of audience dissatisfaction) and conjectures based on studies of the use of a bulletin board system [9, 10], the initial Loops scenario was developed using a prototype consisting of a stack of hand drawn index cards. The initial vision of Loops was akin to a lightweight mailing list that could be started by an ordinary user, and which allowed potential participants control over their degree of involvement in the conversation. The idea was that users could be *aware* of the activities of other participants with respect to the conversation, so that a gathering crowd might entice others to join. Similarly, since this awareness would be shared by all participants and thus enhance accountability, phenomena such as a dispersing crowd might provide a way of shaping a conversation's content, style, or etiquette. Other scenarios and paper prototypes followed, and Loops was widely discussed by members of the lab.

About six weeks after the initial Loops scenario, one of our remote members appeared in the lab with his own prototype. The prototype, which he had named "Babble," was a functioning server and client system. Babble implemented some of the basic Loops concepts, albeit in a form that more closely resembled a combination of chat and bulletin boards than the initial concept's melding of mailing lists and bulletin boards. The adoption of Babble was gradual, taking at least six weeks. Nevertheless, Babble came to play three important roles in the design process. First, it became both the subject and locus of

design discussions. Second, it served as a testbed, with new ideas being tried out both in the original client, as well as in two clients being developed for other platforms (these being part of an effort to infect other, cross-laboratory projects with 'Loops ideas'). Finally, after about ten months of use and design evolution, Babble was deployed to six other groups who, as of this writing, have been using it for about two months.

The Design Rationale of Loops

The design of Loops has been shaped by the following assumptions:

Cues from content- e.g., the ability to see how a conversation has unfolded across time and participants - can enable newcomers to recognize the norms and conventions in a particular conversation, thus enabling them to contribute more effectively.

Social cues - e.g., audience size, who is listening, how actively people are participating - can focus participation and make conversations more engaging.

Relatively small groups, whose members all know one another, or know *of* one another, are most likely to draw on social and content cues as a result of their shared activities and social dynamics.

The first two assumptions are embodied in the design of Loops; the last in the selection of its target audience.

The Conversation as a Single Document

A conversation is represented as a single, persistent document with the oldest items first (e.g., Figure 1). Each comment is preceded by a header which has a time stamp and the participant's name. New comments are appended to the end of the document. This approach is employed by some types of asynchronous bulletin boards.

```
===Friday 12 Dec 97 3:43:44 From: Bill
```

```
Hi Steven!
```

```
===Friday 12 Dec 97 3:44:49 From: Steven
```

```
Helloo Bill. A little guidance please?
```

```
Is the summary we're preparing supposed to be an exercise in feeling good,  
or are we supposed to be giving him hard-headed guidance?
```

```
===Friday 12Dec 97 3:46:55 From: Bill
```

```
yes :-)
```

Figure 1. A segment of conversation. Note here that the utterances in the conversation, including their headers, are shown within a single, non-hierarchical view. Also note that although this conversation segment is synchronous, since the text is persistent, hours, days, or months may elapse between utterances.

As discussed elsewhere [9, 10], this type of representation has a number of advantages both for readers and for writers. Readers, whether they be newcomers or simply infrequent participants, can get an

overview of social norms that govern the conversation by skimming through it. For example, the length of comments is apparent, as is the informality of the conversation (inferable from the simplified syntax and the absence of punctuation), and degree of humor and politeness. By scanning the name and time stamp headers that precede each comment, the tempo of the conversation, the number of participants, and the presence or absence of frequent participants can be inferred. Representing conversation as a single, shared, persistent stream has consequences for authoring as well. The fact that the conversation is persistent and shared increases the potential for accountability. Unlike chat, where conversation is ephemeral, or like mailing lists where the past becomes buried in message archives, accessing the conversation's history is just a matter of scrolling.

Another important element of this representation of the conversation is its single, shared sequential structure: if someone responds to comment A by posting comment B, B will appear immediately after A (except in rare cases when two postings are submitted at about the same time), and that adjacency will be seen by all viewers. This shared sequential structure means that participants can and do participate with short, indexical utterances like "Yes!", "Thank you," and "Great idea!". This type of response is less likely to occur in a mailing list because since there is no shared, sequential structure it is necessary to quote the text being referred to, and because participants are often annoyed at opening a message and finding only an 'insignificant' comment in it. Yet, while such indexical comments may not extend the conceptual boundaries of the conversation, they can make it more convivial and inviting by providing a low overhead way for participants to signal agreement, encouragement, and empathy.

This portrayal of the conversation as a single, sequential document provides a variety of cues for making people aware of the norms and customs in force that are less visible in other, more hierarchical representations of conversation. The conversation's persistence is a boon to asynchronous interaction, and supports accountability as well. However, we also think it is important to provide a synchronous portrayal of social cues.

The Social Proxy

In Loops, synchronous cues for a given conversation are provided by a social proxy, a minimalist graphical representation of users which depicts their presence and their activities *vis à vis* the conversation (Figure 2). This social proxy portrays the conversation as a large circle, and the participants as colored dots, referred to, hereafter, as marbles. Marbles within the circle are involved in the conversation being viewed; marbles outside the circle represent those who are logged on but are in other conversations. The marbles of those who are active in the current conversation, either contributing or 'listening' (that is, interacting with the conversation window via clicks and mouse movements) are shown near the circle's center; with inactivity marbles drift out to the periphery. When people leave the current conversation their marbles move outside the circle; when they enter the conversation, their marbles move into the circle. In our current prototype these activities have optional sonic cues.



Figure 2. A schematic of the social proxy and, to its right, three instances of it. The schematic shows two people (dots 1 & 2) actively involved in a conversation, one inactive person (dot 3), and one person involved in a different conversation (dot 4). Each dot occupies a (virtual) wedge; wedges are created and destroyed as people log on and off. The first instance shows a 'hot' conversation; the second, a dormant one; the third a mixture of activity, idleness, with three people in other conversations.

Although simple, this social proxy gives a sense of the size of the audience, the amount of conversational activity, as well as indicating whether people are gathering or dispersing, and who it is that is coming and going. Also, because the portrayal is visual, it has a perceptual directness (like the glass window) that a list of written names lacks. Experientially, the social proxy is interesting because it focuses attention on the group as a whole, and the coherence (or lack thereof) of its activity.

THE WORKING PROTOTYPE: BABBLE

While stories, scenarios, and rough prototypes are invaluable for getting a quick handle on design ideas, we believe that designing CMC - particularly systems in which social mechanisms play a central role - necessitates moving ideas into a usable prototype as quickly as practical. Thus, having laid out the initial design rationale, we shift our attention to a working prototype called Babble. Initially developed about six weeks into the project, Babble has gone through considerable evolution over the ensuing year. Here we describe the Babble interface as it is now; next, we discuss our experiences from our year-long use of Babble.

Babble, as it has evolved through design and usage, is a CMC system that allows conversation to be threaded (by user-defined topics) and persistent (held on the server until the topic is deleted), and that provides a social proxy that shows the participants and their activities with respect to the conversation (Figure 3). Babble also allows participants to open private, one-to-one chats which are not persistent, so that a completely private channel of communication is available. Written in Smalltalk, it uses TCP/IP and a client-server architecture to transmit both conversation and social information.

A Tour of the Interface.

The Users List

In the upper left pane of the Babble window (Figure 3) is a list of the names of all users who are logged on, each shown with his or her marble (i.e. the colored dots).

The Social Proxy

The upper middle pane contains the social proxy (usually called 'the cookie'), which here shows that all 8 participants are in the current conversation (in this case, the "Commons Area"). Two of these participants have not been recently active; the other six have all 'spoken' or 'listened' (i.e. interacted with the Babble window) recently. Over the course of several minutes of inactivity, a participant's marble drifts towards the periphery.

The Topics List

The upper right pane shows a list of all topics (i.e. conversations), with the currently viewed topic highlighted. Clicking on a topic moves the user to that topic, resulting in the conversation being displayed in the bottom pane of the window, and in the user's marble moving out of the circle (from the

perspective of the other participants). Miniature icons to the left of each topic name indicate how many people are in it (to a maximum of 10), and the topic changes color when new material is added. Topics can be created or deleted by anyone.

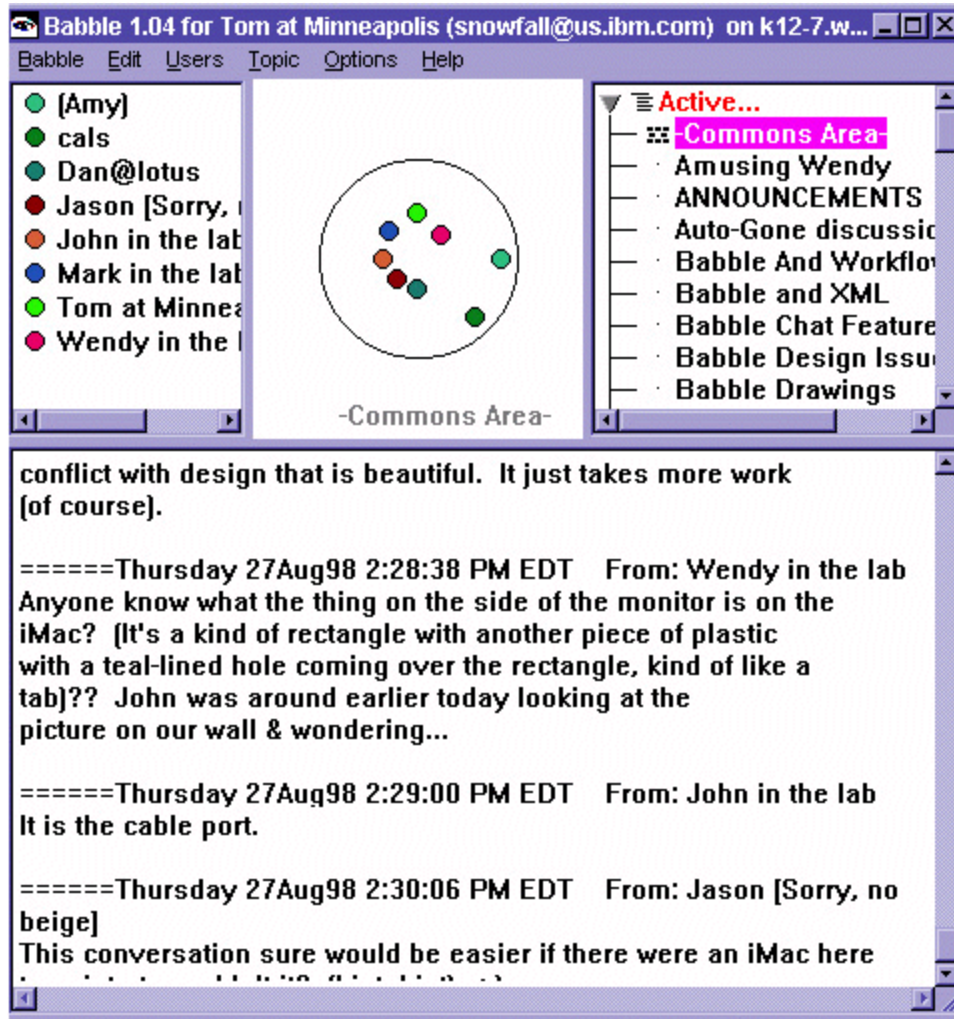


Figure 3. A screenshot of Babble. At the moment shown, all participants are in the same conversation (Commons Area), and most have recently 'spoken' or 'listened'. The two marbles near the periphery of the circle are participants who have not been active recently.

Talking via the Topic Window and Chat

The bottom pane holds the topic's conversation which consists of a shared sequential structure in a single, persistent document. People 'talk' by typing into an entry window; if they select text before beginning, the selected text is 'quoted' and displayed in the entry window. In either case, once the text is composed, the user clicks a "Done" button and the comment is appended to the end of the conversation, with a name and time stamp.

Babble also supports private, one-to-one chat. By right-clicking on a participant's name or marble, a user can initiate a private chat. One experimental feature is that soft key click sounds are transmitted in real time, giving the chat partner cues as to whether and how extensively the chatter is responding; the actual text of the comment is not sent until the chatter clicks "Done." Although chats are not persistent,

participants can, and sometimes do, copy portions of private chats into public conversations.

Other Features

The Babble interface also includes a second, very small window called "the spot" (not shown), which turns green whenever a new message appears in the current conversation. This allows users to minimize the Babble window, using the spot as a monitor for conversational activity while they perform other tasks on their workstations. Clicking on the spot brings up the Babble Window, with new comments temporarily highlighted.

Another feature is the ability to get information about users' activities by right-clicking on their marbles and choosing "Get Info...". This reveals where the user is, and when they were last present in the current topic. This is another way of supporting awareness and accountability.

Babble in Our Lab

Babble began running in our own lab on August 4, 1997. It took 4-6 weeks for the conversational structure that is used today to emerge. That structure consists of a "Commons Area" which is the default entry point, and a set of topics created by Babble users. Figure 4 shows the growth of participants and topics over the first 6 months of use in our lab.

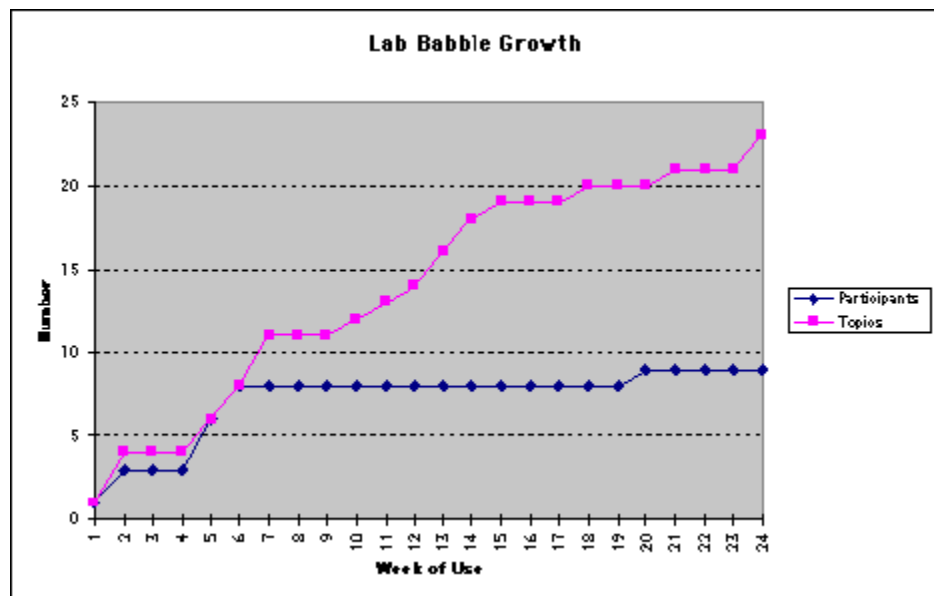


Figure 4. Growth of participants and user-created topics in the first 6 months of Babble use.

The Deployment of Babble to Other Groups

Both because of interest from non-members of the lab, and because of our conviction that designing solely for one's own use is a mistake, we began deploying Babble to six other groups in July of 1998. The groups included two computer science research groups, two other working groups (technical recruiting and marketing groups), and two cohorts (a social cohort of summer interns, and a professional cohort of HCI professionals).

As of this writing (about two months after deployment), three of the deployment groups are using Babble on a daily basis (one CS group, and the two cohorts). Two are making sporadic use of Babble. One group has abandoned Babble after sporadic attempts at use, though the group has so far resisted efforts to "take it away".

DISCUSSION

In this section we will provide a distillation of our experience using Babble. We will primarily draw upon the year long experience of our lab's usage as reported in interviews and visible in the (persistent) Babble conversations and logs. We will occasionally refer to the usage practices in the deployed Babbles derived from observations and interviews, although we believe that the six weeks to eight weeks of their use is too short a period for group behaviors to stabilize (note that in the growth trajectory of the Lab Babble, six to eight weeks of use was really just the beginning).

We are well aware of the drawbacks of reflecting on our own use of our own tool. Two problems, in particular, stand out: first, there is the possibility that we will act in such a way as to fulfill our own expectations; second, there is the possibility that we are overly motivated to use our own technology. In view of the first problem, we will primarily focus on ways in which our use of Babble has diverged from our expectations, as embodied in our initial paper prototypes and scenarios; we will supplement these reports, where appropriate, with observations from the Babbles deployed to other groups. In view of the second problem, note that we are not making claims about the degree of usability of Babble, *per se*, but rather about ways in which it used. Note also that the results of the Babble deployment show that there are no insurmountable usability barriers to its use by other groups.

Opportunistic Interactions

In the initial Loops scenarios, the social proxy was depicted as a way of initiating topic-oriented synchronous chats. People who happened to be in the same topic, it was reasoned, would be likely to have similar intents, and thus be good candidates for spontaneous interactions (also see [16]). While this did indeed occur, it also turned out that the expressiveness of the social proxy triggered more general interactions. That is, when someone's marble moved abruptly (either into the center of circle, or out of the circle), it meant that they were at their machine and that their attention was focused on Babble. These marble movements tended to catch the eye, and thus served as effective triggers for interactions ranging from sociable greetings to work-oriented questions, either via the topic (e.g., as in Figure 1) or in a private chat. Marble movements also triggered phone calls and office drop ins, although we did not track the frequency of these occurrences. In this regard, the social proxy's ability to indicate activity via marble movement and position seems superior to purely textual ways of representing activity.

Group Awareness

Another unexpected effect of using Babble has been its usefulness in maintaining and expanding group awareness. As noted, the social proxy provides synchronous awareness of who is on the system, and who is active. And, obviously, examining the persistent conversation traces (particularly that in the Commons Area) reveals who has been around. What is less obvious is how much awareness comes through the content of the conversation, often as an unintended side effect. Sign offs ('I have to go to the [project] meeting now'), asides ('Dave is right, the Network Nation was published in '78'), questions ("Does anyone know how to do a screen capture"), reveal that one participant is still involved in a particular

project, remind the group that a paper is underway, and suggest that another participant is beginning to document a prototype. Furthermore, the more one knows about the group and its activities, the more that can be inferred from such talk. Because this awareness grows incrementally over days, weeks, and months, and is essentially a side effect of witnessing comments and conversations among other group members, it feels very lightweight. Lab members commented that one of the things they did upon returning from a trip or vacation was to read through the commons area to see what had happened during their absence.

Informality and Sociability

One of our intentions for Loops was that it should bring a more sociable dimension to workplace discussion. We hoped that the persistent sequential representation of conversation would lower the overhead for jokes, puns, thanks, and affirmations, and that the group activity expressed through the social proxy would heighten the awareness of the group as a group. This did appear to happen, in that it provided a venue for sociable talk that - outside of face to face interaction - did not take place via other communication systems.

Similar feelings were expressed in interviews with members of the groups to which Babble was deployed. Our informants told us that they were less careful about punctuation, spelling, and other mechanical aspects of writing when using Babble as compared with email.

"When you are in Babble, it seems like a more relaxed atmosphere and you don't have to watch your spelling, you don't have to have your sentence structure perfect and all that. The [email] system we are using, ... you feel like everything has to be correct because you feel like someone might print out that note and show someone else."

-Recruiter

"I think [Babble is] less formal. I treat it less formal. I wouldn't write mail about someone else's bug unless I check very very carefully that it is indeed in their code. ... It's funny but it's OK to write things that are not 100% finished. ... It's not that thought through ... half-baked ideas are OK. Somehow it's much more like conversation."

-Software Engineer

They saw the (perceived) informality of Babble as an advantage - allowing them to express ideas which were not yet fully formed or to make statements that they were not sure were wholly accurate, without offending others. Their comments are reminiscent of Fanderclai's remarks about MUDs [11]:

The novelty and playfulness inherent in the environment blur the distinctions between work and play, encouraging a freedom that is often more productive and more enjoyable than the more formal exchange of other forums. It is perhaps something like running into your colleagues in the hallway or sitting with them in a cafe; away from the formal meeting rooms and offices and lecture halls, you're free to relax and joke and exchange half-finished theories, building freely on each other's ideas until something new is born.

Babble as a Place

We were taken unawares by the degree to which Babble turned into a place. While we had talked about

how to make Babble more MUD-like (indeed, one topic is called "MOOifying Babble"), we viewed this as future development awaiting the ability to use embedded graphics, rich text and complex page layouts. Nevertheless, Babble came to feel considerably more like a place than we had imagined was possible in a client which permitted only the creation of sequences of (non-rich) text.

One element which made Babble seem place-like had to do with the way different topics developed different feels. In the lab's Babble, several different types of 'places' developed. One was the "Commons Area," a place for people to hang out and talk. The Commons was where most people spent most of their idle time (and for that matter, most of their conversational energy) and became a place for social chit-chat that sometimes segued into work issues ranging from design to administrative announcements. Other topics (e.g. "Babble Problems," "Book Recommendations," "Bad Jokes") were places for general postings with occasional side chat or Q&A. Still yet a different sort of place was the private office or notebook. In early July, one lab member started a topic intended to be, according to its opening comment, "a combination of an on-line office and notebook." The comment continued with "You're welcome to leave me a message, or to comment on things I put here" thus becoming an assertion of ownership and control. This assertion was generally complied with, and the topic became characterized by fairly long essay-like comments by 'the owner,' interspersed with comments from and conversations with 'visitors.' The creation of this topic was soon followed by that of other 'offices'.

The place-like nature of Babble was also entwined with that nature of the language used within it. As noted previously, conversation was typically frank and unguarded. Babble was regarded as a semi-private place. This became apparent when several 'visitors' showed up over a short period (access to Babble requires only possession of a client, a server name, and a port number, and thus may be 'granted' by any current Babble user). One inhabitant of the Lab Babble wrote:

"In the last week or so, D, G, S, and K have shown up. I know the first three, but don't think I know who (or at least which K), K is, and that feels a little weird because to me Babble feels a bit like my office and there are now strangers in it! (Nothing personal, K)."

A similar instance occurred in one of the deployment groups, where a non-group member was invited into Babble. In an interview, one of that group's members said:

"So I think to myself, 'is she listening to every word?' Since it is such a group thing, I would have expected someone to ask the group before inviting someone outside to join..."

Notice that this concern about strangers - those from outside the group's social context - is another manifestation, this time negative, of the accountability supported by the Babble environment.

CONCLUSIONS

Babble is not a bulletin board, a chat system, a MUD, an email system, or a newsgroup. It merges elements of many of these systems, but it is not quite any of them. Babble combines the persistence and shared sequentiality of asynchronous bulletin boards, with the immediacy and informality of MUDS and chat, and presents these along with a perceptually-based social proxy that shows not just those who are speaking but those who are active (i.e. interacting with the interface). In our view, Babble is more akin to a MUD [4] than anything else. However, unlike a MUD where conversation is ephemeral and built objects have persistence, in Babble it is the conversations that persist and the cues that shape interaction (embodied as rooms and objects in MUDS) that are either ephemeral, as in the social proxy, or at least much more tacitly embedded in the persistent conversation

Our experience with Babble suggests that informal, persistent conversation systems fill a communications niche that is currently lacking in many work contexts. Many members of the lab, as well as informants in the deployment groups, felt that having an electronic place including "just the right crowd" was useful for communicating things that they would not communicate in other written forms. We believe that this is one of the most important aspects of Babble: it can be used as a place for unguarded discussion among people who know one another, who understand the contexts within which their remarks are being made. Hyperbole, misattribution, inaccuracy, etc., are a fundamental part of how people talk with one another, and they play an important role: they promote response, and cause people to push ideas farther than they would otherwise. Creative and out-of-the-box thinking arises from playful struggle, from exaggeration, from jumping up and down on top of a soap box, from trying to reconcile contrary ideas, tensions, etc. With an important proviso: All this has to take place in a safe and trusting place.

The notion of a conversational environment as a 'trusted place' is an interesting and challenging one. How - technically, socially, and organizationally - can we balance the need for a safe and trusting place with the organizational imperative to share information? One decision facing us as designers is how and to what extent we "design in" norms and social conventions. For example, if we build in technical mechanisms to provide privacy, in addition to the usability impact, we also eliminate opportunities for participants to show that they may be trusted, or to rely on others to respect their privacy. The Babble prototype has no technical features for controlling access: anyone who has access to the client could, in theory, enter any Babble space. But, because Babble makes users visible (synchronously and, through the "Get Info..." command, asynchronously), this resulted in the group noticing, commenting on, and ultimately discussing how to deal with this issue. We believe that a greater understanding of how to design socially translucent systems which permit social mechanisms to come into play is of great importance in designing CMC and CSCW applications.

Next Steps

We intend to continue deploying Babble to other groups, and studying those deployments as they evolve over time. We expect that groups which adopt Babble will evolve considerably different ways of using it. We hope that a close examination of these cases can provide some insight on how group needs and social dynamics interact with social translucence and the structure of the system to determine practice. We also hope to better understand how to design CMC systems to facilitate adoption in a landscape of shifting institutional needs and practices.

We see a number of future research issues. One obvious issue - since we have considerable difficulty (re)finding valuable nuggets of conversation - is providing tools for structuring conversations (both on the fly and after the fact), as well as tools for navigating, searching, and visualizing large conversations. Another issue is to pursue the development of the social proxy. Our current portrayal, while useful, is extremely simple. We intend to explore ways to make it richer, as well as looking at ways of supporting less synchronous behavior such as recording traces of social behavior over time.

ACKNOWLEDGMENTS

The work described herein is highly collaborative, and we acknowledge the substantive contributions of our Babble colleagues: Jonathan Brezin, Brent Hailpern, Amy Katriel, Cal Swart, and Jason Ellis. We thank our users for their assistance, both tacit and explicit. Thanks to Dave Curbow, Allen Cypher,

Niklas Damiris, Paul Dourish, Jed Harris, Austin Henderson, Charlie Hill, and Shah Xin Wei, Randall Smith, Rachel Bellamy, and John Thomas for great conversations. This paper benefited from comments by five anonymous reviewers and Noboru Iwayama.

REFERENCES

1. Ackerman, M.S. and Starr, B. (1995). Social activity indicators: Interface components for CSCW systems. In *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST '95)*, New York: ACM, pp. 159-168.
2. Bellotti, V. and Bly, S. (1996). Walking Away from the Desktop Computer: Distributed Collaboration in a Product Design Team. *Proceedings of CSCW '96*. ACM Press, 1996, pp. 209-218.
3. Benford, S., Bowers, J., Fahlen, L., Mariani, J., & Rodden, T. Supporting Cooperative Work in Virtual Environments. *The Computer Journal*, Vol 38, No. 1.
4. Bruckman, A. MOOSE Crossing: Construction, Community, and Learning in a Networked Virtual World for Kids. PhD dissertation, Massachusetts Institute of Technology, 1997.
5. Cohen, J. Monitoring Background Activities. *Auditory Display* (ed. G. Kramer). Addison-Wesley, 1994, pp. 439-531.
6. Dieberger, A. (1997). Supporting social navigation on the World Wide Web. *International Journal of Human-Computer Studies*, 46(6), pp. 805-825.
7. Donath, J., Karahalios, K., & Viegas, F. (1999). Visualizing Conversation. *Proceedings of the Thirty-Second Hawai'i International Conference on Systems Science*. (ed. J. F. Nunamaker, Jr. R. H. Sprague, Jr.) IEEE Computer Society Press, 1999.
8. Erickson, T. (1995) Notes on Design Practice: Stories and Prototypes as Catalysts for Communication . In *Scenario-Based Design: Envisioning Work and Technology in System Development*. (ed. J. Carroll). New York: Wiley & Sons, 1995, pp. 37-58.
9. Erickson, T. (1997) Social Interaction on the Net: Virtual Community as Participatory Genre. In *Proceedings of the Thirtieth Hawai'i International Conference on Systems Science*. (ed. J. F. Nunamaker, Jr. R. H. Sprague, Jr.) Vol 6. IEEE Computer Society Press: Los Alamitos, CA, 1997, pp. 23-30.
10. Erickson, T. (1999) Rhyme and Punishment: The Creation and Enforcement of Conventions in an On-Line Participatory Limerick Genre. *Proceedings of the Thirty-Second Hawai'i International Conference on Systems Science*. (ed. J. F. Nunamaker, Jr. R. H. Sprague, Jr.) IEEE Computer Society Press: Los Alamitos, CA, 1999.
11. Fanderclai, T. (1996). "Like Magic, Only Real." L. Cherny and E. Weise (Eds.), *Wired Women: Gender and New Realities in Cyberspace*. Seattle: Seal Press.
12. Goffman, E. (1963) *Behavior in Public Places: Notes on the Social Organization of Gatherings*. Macmillan Publishing Co., New York, 1963.
13. Goffman, E. (1967) *Interaction Ritual*, Anchor Books, New York, 1967.
14. Hill, W., Hollan, J.D., Wroblewski, D., McCandless, T. (1992). Edit wear and read wear text and hypertext. In *Proceedings of ACM CHI '92 Conference on Human Factors in Computing Systems*, pp. 3-9.
15. Hill, W., Stead, L., Rosenstein, M., and Furnas, G. (1995). Recommending and evaluating choices in a virtual community of use. In *Proceedings of ACM CHI '95 Conference*.
16. Isaacs, E.A., Tang, J.C., and Morris, T. (1996). Piazza: A desktop environment supporting impromptu and planned interactions. In M.S. Ackerman (Ed.), *Proceedings of the ACM 1996 Conference on Computer Supported Cooperative Work*, pp. 315-324.
17. Kendon, A. (1990) *Conducting Interaction: Patterns of Behavior in Focused Encounters*.

Cambridge University Press, Cambridge, 1990.

18. O'Day, V.L., Bobrow, D.G., and Shirley, M. (1996). The social-technical design circle. In M.S. Ackerman (Ed.), *Proceedings of the ACM 1996 Conference on Computer Supported Cooperative Work*, pp. 160-169.
19. Pedersen, E. R., & Sokoler, T. AROMA: Abstract Representation of Presence Supporting Mutual Awareness. *Human Factors in Computing Systems: CHI 97 Conference Proceedings*. ACM Press, 1997, pp. 51-58.
20. Roseman, M. and Greenberg, S. (1996). TeamRooms: Network places for collaboration. In M.S. Ackerman (Ed.), *Proceedings of the ACM 1996 Conference on Computer Supported Cooperative Work*, pp. 325-333.