Ubiquitous Access to Unified Messaging: A Study of Usability and the Use of Pervasive Computing

Jennifer Lai
Stella Mitchell
Marisa Viveros
David Wood
IBM T. J. Watson Research

Kwan Min Lee
Stanford University, Stanford, California

The advent of mobile devices and the wireless Internet is having a profound impact on the way people communicate, as well as on the user interaction paradigms used to access information that was traditionally accessible only through visual interfaces. Applications for mobile devices entail the integration of various data sources optimized for delivery to limited hardware resources and intermittently connected devices through wireless networks. Although telephone interfaces arise as one of the most prominent pervasive applications, they present interaction challenges such as the augmentation of speech recognition through natural language (NL) understanding and high-quality text-to-speech conversion. This article presents an experience in building an automated assistant that is natural to use and could become an alternative to a human assistant. The Mobile Assistant (MA) can read e-mail messages, book appointments, take phone messages, and provide access to personal-organizer information. Key components are a conversational interface, enterprise integration, and notifications tailored to user preferences. The focus of the research has been on supporting the pressing communication needs of mobile workers and overcoming technological hurdles such as achieving high accuracy speech recognition in noisy environments, NL understanding, and optimal message presentation on a variety of devices and modalities. The article outlines findings from the 2 broad field trials and lessons learned regarding the support of mobile workers with pervasive computing devices and emerging technologies.

Requests for reprints should be sent to Jennifer Lai, IBM T. J. Watson Research Center, 19 Skyline Drive, Hawthorne, New York 10532. E-mail: lai@watson.ibm.com

1. INTRODUCTION AND MOTIVATION

"To date, the way we interact with computers has been incredibly unimaginative and limited" (Norman, 2001). One of the goals of researchers in the field of pervasive computing has been to improve on the current model of interaction with computers. Pervasive computing deals with computers that are accessible anywhere and anytime, as well as computers that are embedded in other devices and are no longer necessarily recognizable as a computer. The plethora of mobile devices created in the past few years, coupled with the increased bandwidth of the mobile Internet, address the ever-growing need for pervasive communication among people. Perhaps the most ubiquitous device today is the telephone. Given the greater mobility of workers everywhere, both manufacturers and designers are stepping up to the challenge of giving users access to all the information they have when seated in their offices. It would seem that the term *mobile worker* no longer describes a transient state for a person traveling to the office, but often represents a category of corporate employees whose only designated work location is the one they have in their home or car.

Creating a highly usable interface for accessing e-mail messages over the phone represents a substantial challenge. It is not unusual for the average knowledge worker to receive over 100 messages per day. The telephone keypad access model that was established for voice mail messages does not work when applied to large amounts of textual information. When viewing the daily onslaught of messages in a Graphical User Interface (GUI) setting we routinely do a visual triage, scanning for messages from people who are important to us, or for message topics that pique our interest. This triage is difficult to do when relying on auditory input, which is slower than the visual channel. Although most people can produce speech easily and quickly, they cannot listen nearly as easily and quickly. Consider the following input rates (Schmandt, 1994): listening at 175 to 225 words per minute (wpm); reading rate at 350 to 500 wpm.

Using speech technology to interact with computers when there is no keyboard or monitor available continues to garner a lot of interest. Although users in some parts of the world (e.g., Japan and the Philippines) are more accepting of the challenge of inputting text via the telephone keypad, many users find this form of input too cumbersome. In addition to facilitating input, speech also provides the necessary extra channel when the task requires users to have their hands and eyes busy. Because people can look and listen at the same time, speech can provide an alternate method of feedback when driving a car or working on the factory floor. One of the most important features of a speech application is the human factors aspect. Like any human interface technology, the success of the product depends on the acceptance by the people who will use it. Speech recognition must offer benefits beyond novelty. There must be an explicit reason to include speech technologies in the application. This demands that one understand not only the technology but also the task that the application is supporting and the users whom will be using it.

Applications for mobile devices entail the integration of various data sources optimized for delivery to limited hardware resources and have to deal with intermittent connections through wireless networks. Many of today's limits on perva-

sive computing reflect barriers that will not exist in several years. As designers and developers working in the field today we have to work around or run head-on into issues such as gaps in network coverage, imperfect recognition technologies, and cumbersome devices. This article reports on our experience in this challenging arena of pervasive computing, building an automated assistant that might one day be used as an alternative to a human assistant.

2. PRIOR WORK

Ubiquitous access to personal and business information has been getting a lot of attention the past decade. Due to the increasingly mobile nature of work and the large amounts of personal and business communication in everyday lives, delivering information on personal digital assistants (PDA), pagers, or cell phones is gaining interest from both academic and business communities. A number of commercial systems provide speech-based ubiquitous access to information. One of the earliest products in this domain is from Wildfire Communications (www.wildfire.com). Wildfire Communications gives mobile users voice access to address book entries and voice dialing capabilities. Other commercial ventures in this space include TellMe (www.TellMe.com), AOL (www.aol.com), General Magic (www.genmagic.com), HeyAnita (www.heyanita.com), and Webley (www.webley.com). These providers have voice-enabled access to e-mail messages, calendar information, address book data, movie reviews, restaurant listings, and stock prices. The AOL By Phone service is moving voice access to information into the mainstream with over 200,000 subscribers and more than one million calls.

Several university research projects are looking further out to the day when pervasive computing devices will be truly ubiquitous and always available, so that appropriate information and services are at the users' fingertips. These include the InfoSphere project (www.cc.gatech.edu/projects/infosphere; a collaboration between Georgia Tech and the Oregon Graduate Institute); the Oxygen project at MIT (Dertouzos, 1999); the Portolano project at the University of Washington (Esler, Hightower, Anderson, & Borriello, 1999); and the Endeavour project (http://endeavour.cs.berkeley.edu) at the University of California, Berkeley. A common theme among these projects is that devices will be specialized and transparent to users, available at the point of need with the necessary function, using interfaces that are well suited to humans. Oxygen places a strong emphasis on the use of speech in the interface so that team collaboration might be just a shout away. As part of Endeavour project, the Iceberg architecture (Wang et al., 2000) supports ubiquitous access to a myriad of data types and services across a range of devices and interfaces. Iceberg has an open service architecture and is implemented on current telephony and data networks. An initial proof point for their architecture is a Universal Inbox (Raman, Katz, & Joseph, 2000) capturing various data and delivering them through the proper network to the desired device according to a set of user preferences.

Researchers at the MIT Media Lab's Speech Interface group identify four important issues to consider with voice access to ubiquitous information—minimizing in-

terruption, user adaptation, location awareness, and user interfaces (Schmandt, Marmasse, Marti, Sawhney, & Wheeler, 2000). To minimize unwanted distraction, the service needs to filter and prevent less important messages from being delivered at inappropriate times. The service needs to adapt to changes in user activities so that messages can be sent to the correct device in a timely manner. Further, the service should infer the location of users and provide location specific information (e.g., grocery list when a user is near a supermarket). Last are the interface issues. Voice interfaces, which often are well suited to mobile situations, where the user's eyes and hands are engaged in other activities, pose some unique challenges (Kamm & Helander, 1997). Of key importance are accuracy and understanding of the input, as well as natural language (NL) generation and large document presentation on output. To facilitate addressing these issues, the speech group at MIT has developed the Galaxy (Seneff, Lau, & Polifroni, 1999) distributed architecture to address the complexities associated with developing pervasive voice-user interfaces.

3. USER POPULATION AND REQUIREMENTS

Our MA prototype was piloted with two user groups. The initial pilot was conducted with IBM employees from the Global Services organization, the second with employees from the Research organization. We first describe the user population within Global Services, because many of the same characteristics of the first user group are also true of the second group. The solution was defined based on the requirements and characteristics of the initial user population, who are representative of other mobile knowledge workers.

The first user group consisted of client service leaders (CSLs). These executives are assigned to a single client and are responsible for overseeing all the various engagements (e.g., software and hardware sales, consulting services) that IBM has active with the account. This requires the CSL to interact with somewhere between 10 and 15 other IBM team members from different organizations who are working with the account, as well as being the primary interface between IBM and the client.

We followed a standard user-centered design methodology to define and validate user requirements, conducting interviews with nine different CSL executives; six within the United States and three overseas. The interviews with the client leaders in Asia and Europe were conducted after the ones in the United States to determine if there were significant variations in the requirements that might be due to geographic differences. Interviews were conducted face-to-face when possible; but to avoid overseas travel, the European and Asia interviews took place over the phone. All the interviews consisted of 1 hr of exploratory, open-ended questions, as well as follow-up questions (to clarify points of discussion raised by the users). The findings from the interviews showed a series of both challenges and opportunities involved in providing a telephony-based solution to the selected user group.

When queried as to their current cell phone usage, we found that, although each of them owns a cell phone, the current usage was not very high. There were several reasons given for this. First are concerns about the cost of the calls and a desire to not exceed the allocated number of minutes in their plan. Second, users found the

quality of the call was often low due to problems with cell coverage, and thus would seek out a landline when they needed to call the client or be on the phone for an extended period (e.g., conference call). In addition, interviewees said that they keep their phone turned off while in meetings (which represents a good portion of the day), and thus are not reachable during these times. Other reasons cited for low usage were security concerns, lack of signal due to poor coverage, and a small number of entry numbers stored in the phone's address book. Given the users' reluctance to incur additional minute charges we knew that the interaction would have to be efficient in its presentation of key information.

All but one of the users interviewed have secretaries (usually supporting more than one executive). The assistant often serves as a focal point for communications because they know the whereabouts of the CSL, as well as the best means to contact him or her. They also take care of most of the calendar tasks, as well as booking travel plans and filing expense statements. In some cases, the secretaries monitor the executive's e-mail for important messages. If the CSL carries a pager (not all do), the secretaries occasionally send pages with calendar updates, reminders of conference calls, or other timely information. Providing a software assistant when the user already has a personal assistant can sometimes be viewed as superfluous. The solution would have to be designed to be complementary to an assistant's services, as well as being able to provide many of the same services for other users who do not have assistants.

It quickly became clear to us, given their high levels of responsibility, that this is not a user group with a large degree of tolerance for imperfect or emerging technology solutions. In response to one of the open-ended interview questions ("What would you describe as the greatest daily challenge in your job?"), one user replied, "I have a quota of half million dollars a day. Can your technology help me make that quota?" These are high-pressure jobs, where the user population is time challenged and information overloaded. They did not want "yet one more device" to carry around with them. Most feel they already have too many, and any new device would have to be a consolidation or replacement of one or more of their existing devices.

However, the fact that this user group is time challenged also represented a good opportunity for us to help support their daily work routine with a pervasive computing solution. These executives are a critical link within the team of IBMers that provide services and sales to an account. They often need to be reached for pricing approval or need to reach others for answers to questions the client has raised. Their communication needs are often urgent, and responsiveness can mean the difference between making a deal or losing it. The CSL and many of the other team members that make up the dynamic client team are usually mobile. Most have at least two work locations that they use, and they are often on the road. When they are in the office, they need to be on conference calls for hours at a time, and thus are not reachable except through their secretaries or by instant-messaging systems.

Another critical aspect of the solution requirement that was discovered through the interview process was the inclusion of other key team members. The prototype solution that was to be created for the CSLs needed to work well for the other team members assigned to the same account. One executive mentioned, "what's good

for the goose is good for the gander." The CSLs did not want to be singled out for receiving a cool new "toy," and they wanted to ensure that the solution would solve a problem that was meaningful for all the team members.

4. THE MA: SOLUTION DESCRIPTION

Based on these findings, a set of requirements emerged for these mobile workers. The focus of the solution was one of improving communications within the team and supporting increased responsiveness. Given the need for only certain people to get through to the users when they are mobile and otherwise occupied, we decided to place a software assistant in charge of answering the phones, determining the identity of the caller, and tracking down the user if the caller was identified as being on the "key person" list for that user. If the caller opted to leave a voice mail message instead, the message showed up in the user's inbox as an audio recording. Having the voice mail messages appear in the inbox allowed users, who were otherwise unavailable due to lengthy conference calls, to become aware of who was trying to reach them and what the nature of the call was about.

Users were notified of the arrival of voice mail, urgent e-mail messages, and changes to the current day's calendar through text notifications. These are sent as text messages to the user's e-mail addressable cell phone. The alert contains enough information to be useful (e.g., the sender name and subject of the message) and to allow the user to decide if any immediate action needs to be taken (e.g., call in and listen to the full text of the message). A major component of the solution involves giving users ubiquitous access to unified messages (e-mail, voice mail, and fax). Calendar information and messages can be accessed in one of three modalities:

1. From a desktop computer in a combination of audio and visual modes. This is similar to the standard configuration today, except that voice mail is received as an audio attachment to an e-mail message and can be listened to at the desktop (see Figure 1). For internal calls, the identity of the caller is listed in the header information of the message. E-mail messages created using the MA system are also received as audio attachments.
2. From a SmartPhone (a cell phone with a four-line display and a Web browser) in a silent visual mode. In this case, users read their e-mail and calendar entries on the phone's display once connected to the network. Lotus's Wireless Domino Access 1.1 product was used to deliver this functionality. Although reading text on such a small display can be cumbersome, the primary advantage of using the SmartPhone for visual output is that the interaction can take place in total silence. This is advantageous when privacy is a concern or there is a wish to not disturb others.
3. From any telephone using speech technologies (recognition and synthesis) in an auditory mode. In this case users speak their requests for information, which are interpreted by the MA. Examples of requests include, "Do I have any messages from John?," "What's on my calendar next Friday at 10:00 a.m.?," or "Play my voice mail messages." The MA replies to their queries using synthetic speech and reads them the requested messages or calendar entries. Text notifications of the arrival of

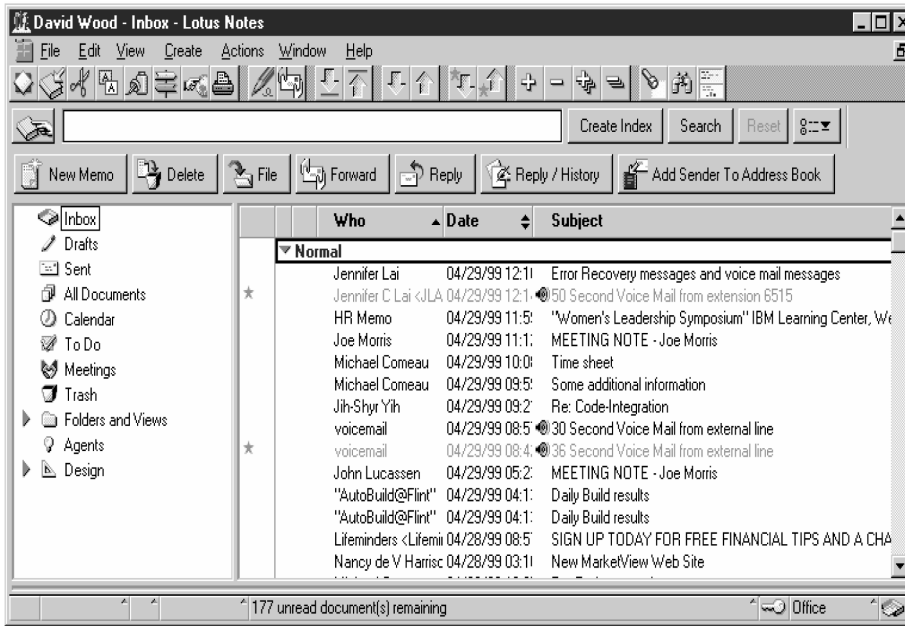

FIGURE 1 Sample inbox with voicemail messages.

urgent e-mail messages and voice mail can be sent to the SmartPhone or any e-mail addressable device.

5. THE CONVERSATIONAL INTERFACE

Unlike grammar-based systems (Schmandt, 1984), which require the user to speak one of a predefined set of sentences, NL systems act on unconstrained speech. The basic components of our NL system (Davies et al., 1999) consists of a speech recognition engine, dialog engine, and text-to-speech (TTS) for output. See Figure 2 for a high-level chart of the components of a conversational interface.

5.1. Speech Recognition

Speech recognition technology, or speech-to-text, maps spoken input into text. We used IBM ViaVoice (V.9.0) with a speaker independent model. Although we initially bootstrapped the recognition models with domain grammars, the data collected from Wizard of Oz sessions with users proved to be the most valuable for improving recognition accuracy. The training data set consisted of approximately 153,000 sentences, 949 distinct words, and 33 embedded grammars. Thirty of the grammars are static; and 3, which contain people's names, are created dynamically when the system is invoked. Given the circumstances of usage (mobile users), highly accurate speech recognition is challenging due to poor signal from cellular telephones and background noise. This challenge is further compounded by the

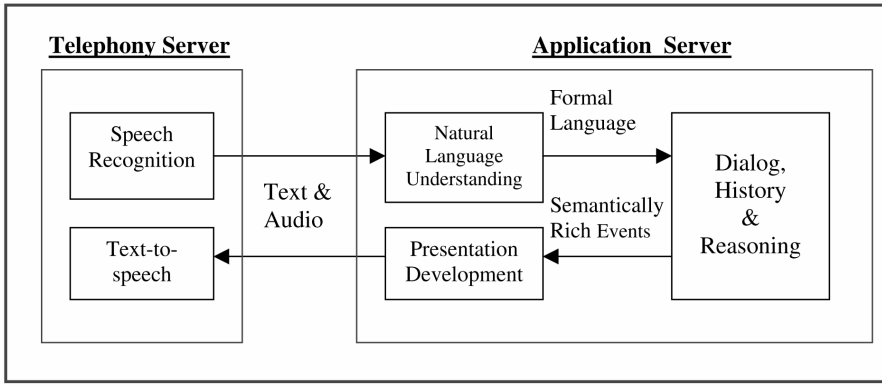

FIGURE 2 Components of a conversational interface.

usual accuracy problems associated with speaker-independent systems, such as users speaking English with foreign accents.

5.2. Dialog

Our NL technology maps a user request into a formal language statement. For example, the request, "Do I have any voice mail from John?" is mapped to *QueryMessages* (type = phone, sender = John). We used the IBM ViaVoice Telephony Toolkit 1.2 NL engine and a custom statistical domain model to perform this mapping. Our forms-based approach to dialog consists of a set of forms, one for each function (e.g., QueryMessages). Each form has one or more slots for user input and defines the set of attributes (e.g., sender, date, start time) that can be used for an operation. Every slot has corresponding system prompts and input validation procedures. If the user has uttered an incomplete request, information can be solicited from the user using these slot-level prompts. When enough slots are filled, the form-level function is initiated. Our current system uses 24 forms.

Dealing with subtlety and ambiguity in user requests is challenging. Consider the phrases, "Send *a* message to John" (user wants to create a new message) versus "Send *the* message to John" (user is forwarding a specific message). Although our NL models can easily distinguish between these requests, it is difficult for a speech recognizer to make this distinction, particularly in a noisy environment. Also, NL does not always deal well with ambiguous user request. For example, if the user says "new mail," it is not clear if the intention is to compose a new message or to check for new messages. To date, the frequency of these types of problems has been low, and we have deferred dealing with this problem for the time being.

5.3. TTS

Although many commercial systems use recorded human speech for the fixed prompts (e.g., "Welcome. Please select e-mail, calendar, or reminders") and synthesized speech for the dynamic text, we chose to use TTS (ViaVoice V. 5.0) for all the

system prompts. This was partly due to a need for expediency and partly as a nod to research that indicates that comprehension is facilitated when the voice is consistent (Gong & Lai, 2001).

6. DESIGN DECISIONS

Our goal for the voice interface is to simulate human–human conversation. We want users to interact with the system in a way that is similar to interacting with a human assistant. The hope is that inexperienced users can interact with the system without great difficulty or a lot of training. One of the biggest problems with telephone-based speech interfaces is letting users know what they can say to the system (Yankelovich, 1996). Speech-only interfaces are in some ways similar to the old command line interfaces, where the set of acceptable commands needs to be known in advance. The GUI was developed to overcome this problem, and user choices can be presented with menus, dialog boxes, push buttons, radio buttons, or check boxes. This box of tools disappears with speech-based telephone interfaces, and the resulting design challenge is substantial. We hoped to overcome this challenge with the use of NL technology, carefully crafted system prompts, adequate auditory feedback to the users, and an interaction style that supports random access to messages.

6.1. Prompt Design

Most speech-based telephone interfaces on the market today use a style of system prompts that create a directed dialog. As the name would suggest, the interaction is controlled by the system, with the system "directing" the user what to say. An example of a directed system prompt is, "Welcome to the U.S. Airline Travel Information system. Would you like information about an existing or a new reservation? Please say, existing reservation or new reservation." The reason most systems use this style of interaction is to increase the accuracy of the speech recognition. By telling the user what to say, the chances of the user speaking an out-of-grammar utterance are much lower. A grammar is the complete set of all possible legal sentences that a user can say at any given moment of the interaction (Ballentine & Morgan, 1999). Grammar-based systems can only "hear" sentences that are defined in the grammar. The disadvantage of directed dialogs is that the user can sometimes feel handcuffed, confined to the passive role of waiting for the system to request the next piece of information. Much of the naturalness of speech is undermined when the interaction relies too heavily on the use of directed dialogs. With NL technology, the style of interaction can be more conversational without introducing a high degree of error. Therefore, our ready-state system prompts are more open ended (e.g., "I'm ready to help") than they would be with a grammar-based system. To reduce repetitiveness, each time the system is ready to receive the next user request, we use variations of "What now?" such as, "What else?," "How can I help?," or "What's next?"

System: Hello, please tell me your name.

Caller: Sally Smith.

System: Welcome Sally. Please enter your password.

Caller: 1 2 3 4 5 6.

System: Good job Sally. Please wait while I retrieve your messages….

You have 23 new email messages and 2 new voicemail messages since your last call. I'm ready to help.

Caller: Do I have any message from George Grant?

System: You have no messages from George Grant. What else?

Caller: List my messages.

System: In the first set of 4 messages, the first message is from Peter Poncet and is about "Update" , the next is from Alice Newton about "Bob's retirement", the third is from Diane West about "Friday's Meeting", and the fourth is from Larry London about "Updates Due Monday". What now?

Caller: Play the third message.

System: The third message was received today at 10:50 am from Diane West. The message reads ….

Caller: What's on my calendar next Friday?

System: On Friday June 22nd you have two entries. The first is from 10 am to 11:30 am with subject …

Caller: Create a one hour meeting at 9 am.

System: What is the subject of the meeting?

Caller: Meet with Diane West regarding NY Times article.

System: A calendar entry was created for Friday June 22 at 9 am. What next?

FIGURE 3 Sample mobile assistant conversation.

Although turn taking in human conversations (Clark, 1993) is often characterized by silence (i.e., when the first speaker is done and ready for the second speaker to speak, he will remain silent), we opted for a direct cue more reminiscent perhaps of the style of conversation adopted by two users of a walkie talkie. Because those conversations are more at risk, given the two modes for a walkie talkie (listen or speak), these users adopt use of the word *over* to indicate an end of turn. We similarly use a chime at the end of the system prompt to signal that the system is done speaking and is ready to listen. Although many commercial systems have opted to remove the beep and to rely on silence for turn taking, the chime was not something our users complained about. Perhaps because so many of the calls to the MA system are made on cell phones, and these interactions are more strained than face-to-face interactions due to varying degrees of cell coverage, the chime was not perceived as interfering with the interaction. A typical conversation with the MA is shown in Figure 3.

6.2. Random Access to Messages

We have already mentioned that presenting large amounts of textual data using only speech presents a considerable design challenge. User questionnaire data showed that our users receive a minimum of 70 messages per day. When using a vi-

Table 1: Mapping of Voice Functions to Visual Scanning Functions

Variables	*Function*	*Implemented*
Overview	Summarize that message	Yes
	What's in my inbox?	Yes
	List message header information	Yes
Zoom	Any messages from *the team*? (a named group)	No
	Any messages from Marisa Viveros? (a specific person)	Yes
	Any messages about the manuscript? (a specific topic)	No
Filter	Don't read me any more messages like that one	No
Details	Play first urgent e-mail	Yes
	Read me the message from Mike	Yes

sual display, users engage what can be thought of as the visual-information principle, "overview first, zoom and filter, then details on demand" (Shneiderman, 1998) to sift through newly arrived messages. According to this principle, users would first gain an overview of what is in their inbox (e.g., approximately how many new e-mail and voice mail messages have arrived), then zoom in on items of interest (e.g., there are two messages from my boss, and three about today's meeting), filter out the uninteresting items (e.g., there are a bunch of messages from the Listserv), and finish with details on demand (e.g., read the first message from the boss). There are several functions that we incorporate into the spoken interaction to facilitate these same four steps. See Table 1 for examples of voice browsing functions mapped to the four steps.

6.3. Feedback

In a voice-only system, where the user is figuratively blind when interacting with his or her data, feedback is critical to the success of the interaction. This is especially true given that both speech recognition and natural language understanding are prone to errors and misunderstandings, which can cause the wrong action to be taken by the system. However, the designer must also keep in mind that speech is a slow output channel, so feedback needs to be sufficient but not overly verbose. We use both explicit and implicit forms of confirmation to feedback to the user what the system is actually doing. In the case of queries, we reiterate the attributes provided by the user. For example, in response to the request, "What's on my calendar tomorrow afternoon?," the system replies, "Tomorrow between 12:00 p.m. and 6:00 p.m., you have 3 items on the calendar. The first is … ." In the case of functions that are difficult or impossible to undo (such as sending an e-mail message or deleting one), we use an explicit confirmation. If the user asks to "delete the message," the system will prompt, "Are you sure you want to mark the message from Jane Doe about Tomorrow's Meeting for deletion?" Although this prompt might seem longer than it needs to be, it contains two pieces of information that are critical to the user. First it uniquely identifies the message that is being deleted (by sender and subject). Second, it informs the user that the message is only being marked for dele-

Table 2: List of Telephone Keypad Shortcuts (DTMF) and Their Function

DTMF	*Function*	*DTMF*	*Function*
0	Cancel the current request	6	Delete the message
1	Play the e-mail message	7	Repeat the message
2	List e-mail headers in sets of four	8	Reply to the message
3	Play phonemail message	9	Forward the message
4	List today's calendar	*	Interrupt the spoken system output
5	Record a greeting	#	Next (day or message)

tion in the Lotus Notes inbox. It is not deleted until the user manually refreshes the inbox at the desktop. This allows the user to make changes if a message was inadvertently marked for deletion.

6.4. Telephone Keypad Shortcuts

Despite the potential richness of the conversational interface, we found that many users requested telephone keypad (also referred to as DTMF) shortcuts to frequently used functions. This was to support their needs for privacy (if using the system in a public setting), as well as to be considerate to the people around them. The DTMF input is also very useful when traveling through areas with low cell phone signal, because a distorted acoustic signal negatively impacts the accuracy of the speech recognition. The mapping of keys is shown in Table 2. The existing set covers all the commonly used functions within the application. As the application grows, we believe that a static mapping will be insufficient, and contextual information will need to be used in conjunction with the key to determine the nature of the request. For example, if a confirmation has just been presented to the user, keys 1 and 2 could be remapped to *yes* and *no,* respectively. Or, if the user has just listened to a list of five e-mail message headers, keys 1 through 5 could correspond to messages in that list. The difficult part of remapping the keys based on context is getting the user to understand and remember which key can be used for what.

7. INITIAL FIELD TRIAL

The initial field trial started at the end of September 2000 and concluded in March 2001. Users were included in the trial as part of a team. Each team was lead by a CSL, who worked with us to define which team members would be included, because we limited each team to 10 people. Users were given a Web-enabled cell phone for text alerts as well as access to messages and calendar data. The user can decide based on his current context if a hands-free voice interface or a silent text interface is best suited for the moment. Users in this pilot were also given a PDA. The PDA was used for instant messaging (IM) to team members by means of the IBM SameTime server. When a user is mobile and in the field, it is not always possible to contact other team members even when they are in the office. This may be because the office-based person is on an extended telephone

call (e.g., conference call), has the phone in "do not disturb" mode, or is in a meeting and thus away from the phone. However, most of our users have IM software running on their laptops and the buildings are setup for wireless network access. Therefore, a team member who is in the field can always reach a colleague by IM. The office-based team member can respond by IM, even if currently on the phone, or can send e-mail with any requested information.

7.1. Outcomes

When the field trial concluded in March 2001, we had gathered a lot of good data about the particular challenges of delivering business e-mail over the phone as well as a large body of spoken utterances to refine our recognition and NL models. From January 1, 2001 to March 7, 2001 (when the trial ended) users accessed the MA solution an average of 2.5 times per day. The average length of the call was 5.49 min. It is interesting to note that although the CSLs are highly mobile, not all the team members are. Several of the IBM employees who are working with an account are actually physically located on-site at the client location. They have connections for laptop computers and are hardly mobile at all. When calling the MA to try various functions that we asked them to try, they often were looking at their e-mail on their laptops while interacting with the voice system.

Some of the challenges that we faced have to do with the rich and diverse nature of e-mail messages. E-mail has become not only a tool for communicating but also for sending attachments, calendar invitations, and complex histories of messages. Our users told us that sometimes just knowing that a presentation or Word document has been sent to them was not enough. Even though we provide the ability to print the attachment on a fax machine, the user really just needs to work on the attachment and a voice interface can not get him there yet. Dealing with calendar invitations (e.g., accepting, rejecting, proposing an alternate time) is a complex function that we have not yet added to the MA. Last, parsing what is contained in the e-mail message and making a reasonable presentation by voice is a substantial challenge. Consider messages that include long histories of forwarded messages and replies. Often when a user is reading his e-mail in a GUI setting, he will have to scroll to the bottom of the note to begin to make sense of what he has received. Other times, the history of the message is known to the user and just the few words added to the top of the note is all he is interested in. Being able to present all the information with synthetic speech, in a concise manner, with error-free navigation was a task that we sometimes fell short of.

For users who are highly mobile and who rely on their assistants to maintain their calendars, they found the ability to make a quick call to the MA to hear the day's calendar entries to be invaluable. Several mentioned that when they leave for work in the morning they are not always sure where the first meeting of the day is, nor even what time it is scheduled for. A questionnaire at the end of the trial probed users for both qualitative and quantitative feedback on their experiences with the MA. The average rating for the usefulness of the voice-based access to messages was 3.2 on a scale of 1 to 5 (with 5 being the highest). Some of the less mobile users commented that they would have given an even higher rating if they were traveling a lot. One thing

that became apparent was that the MA was new and different enough to provoke questions and curiosity from clients who witnessed IBM executives accessing their information on a phone and speaking to a virtual assistant. One user commented, "the MA ranks especially high on the cool factor." Other users found the ability to send an IM to a coworker while at client locations very helpful. A representative comment was, "I liked the ability to surreptitiously contact a resource during a meeting to get a quick answer without interrupting the meeting or losing momentum on the point of discussion." Last, user comments seem to indicate that the MA was facilitating the ability for team members to be responsive to each other: "We are just entering (trial) but already members of the team who are in Seattle, Denver, and New Jersey are experiencing significant benefits in our ability to communicate."

8. SECOND FIELD TRIAL

The second user trial was conducted at IBM T. J. Watson Research Center, initially with employees who are second line managers and above. Like the CSLs, these users were identified as being likely to need a pervasive computing solution due to their high level of mobility and critical communication needs. Unlike the first trial, which ultimately had four teams of 10 people, the second trial did not involve communication within a team. Users were enabled individually as opposed to part of a group. The second trial involves a much larger group of users (over 250 people), because its scope was expanded to include first-line managers as well as nonmanagers. Data collected from the first field trial was used to retrain the recognition and NL models thereby enhancing the end-to-end accuracy for the second trial.

8.1. Outcomes

We examined the usage data from the much larger subscriber base for a better understanding of when and why the MA was being used. The data was analyzed for the time period from June 20, 2001 to January 12, 2002. Figure 4 shows the average number of calls to the MA per week, per user. It shows that after an initial period of exploration and discovery, usage settles down to an average of .097 calls to the sys-

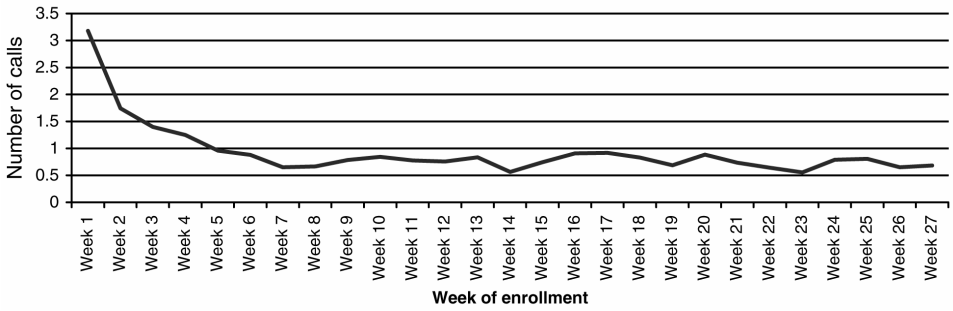

FIGURE 4 Average number of calls per week per user.

tem per week. This is a clear indication that not all users see the need to call daily and that usage is dependent on circumstances. Many users who are not highly mobile do not use the voice system to check messages on a daily basis and perhaps only call on Sundays, when stuck in traffic, or when they can not remember where their first meeting of the day is taking place. Others, who spend more time in meetings away from their laptops or on the road, call in to check more frequently. One user commented that his usage had been low for several weeks because a lull in his schedule found him in his office, sitting in front of his computer every day. Given these circumstances, he did not feel the need to check messages by telephone. It should be noted that the average number of calls to a system like this is expected to be lower than for a traditional voice mail system because users can listen to their voice mail from their desktop computer.

Figure 5 shows the average function breakdown per call. Because every major function can be invoked with either a spoken request or a DTMF key, Figure 5 also shows the distribution of usage between these two modalities. However, because the DTMF key mapping was only made available later in the trial, and not all users are equally aware of the functions, this data is still inconclusive. Functions are listed in order of frequency, with the 'other" category grouping all the lesser used functions.

There are several items of interest in Figure 5. First is the dominance of the "play" function. This function is engaged any time a user asks to hear a specific message (e.g., "Play the first message" or "read the message from Jim"). Although many users mentioned the value of hearing the current day's calendar read to them in the morning, it seems that the system's primary use is to play messages. The second most frequently used function is the interrupt key. Although many speech systems have a spoken barge-in function, our system currently requires the use of a DTMF key to interrupt the system when it is speaking. The high frequency use of the interrupt key may be an indication that the system occasionally plays messages that the user is not interested in hearing, or merely that users can often determine what they need to know from a message without listening to the whole thing. The

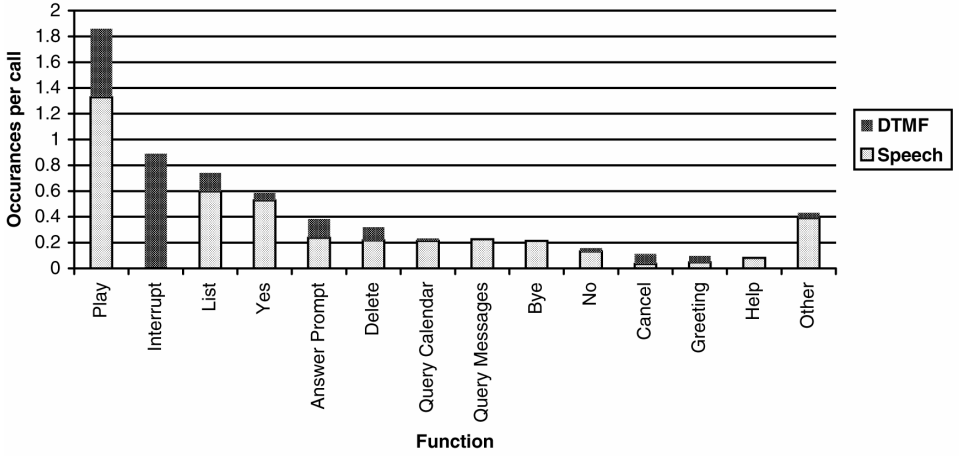

FIGURE 5 Average functional composition per call.

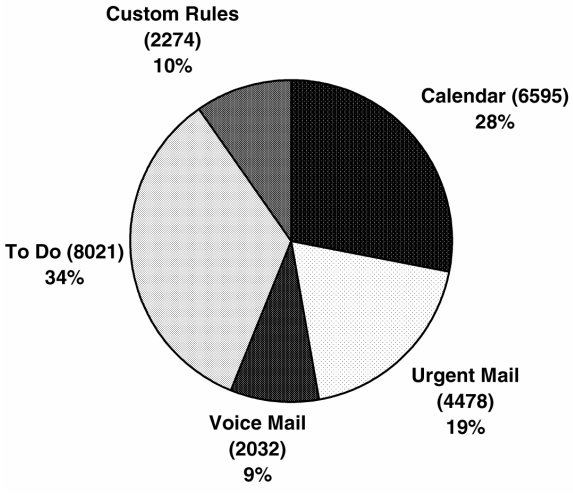

FIGURE 6 Distribution of alerts by type of notification.

"list" function allows users to hear the header information (i.e., sender and subject) of the messages in sets of five messages at a time. If any of the messages are of interest, the user can listen to them by indexing into the list (e.g., "Let me hear the fourth message"). The other primary method of navigating the inbox is to listen to the first message, then interrupt the reading of the message if it is of no interest and continue to the next message. Although we gave users several options to support random access into the inbox (e.g., "Do I have any messages from Jennifer Lai?"), we found from examining the usage logs that many users still navigate sequentially through their messages. Whether this was because they were unfamiliar with the other options, or from a desire to be thorough, is unclear at this time.

Also of note from Figure 5 is the high ranking (among the top 13 functions) of the "yes" and "bye" functions. The yes function is engaged anytime a user responds positively to a confirmation dialog (e.g., "Are you sure you want to send this message to Jim Brown?"). The use of the bye function was interesting to us because there is actually no need for users to engage in the social "nicety" of saying goodbye to the system. A simple hang-up suffices. However, when the system detects that a user has spoken some form of goodbye, it replies with an affable statement along the lines of, "Thanks for calling. I will be waiting for your next call." A plausible explanation for this behavior on the part of our users can be found in the "Computers are Social Actors" paradigm (Reeves & Nass, 1996), which states that humans engage in many of the same social responses with computers as they do with other humans (without being fully conscious of this behavior). However, another possible explanation is that users like the sense of closure they get when indicating their intention to hang up and having the system acknowledge that intention. This could indicate that they have no transactions pending or unresolved requests.

Analysis of usage data showed an overall recognition accuracy of 86.22%. We have a speaker-independent system with no current means to create personalized enrollment data. Enrollment data is normally used in speaker-dependent or adaptive systems to improve recognition rates when users speak with a heavy accent. If we do not include non-native English speakers, the accuracy increases to 88.33%.

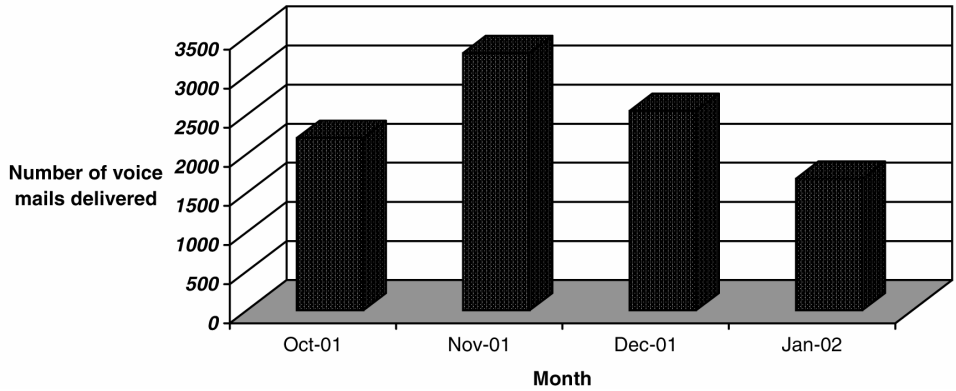

FIGURE 7 Number of voice mail messages delivered by month (through January 12th).

The recognition accuracy would be higher if we did not have a large percentage of the calls made to the system on cell phones, or if were using a grammar-based system.

Figure 6 shows the distribution of the alerts sent by the MA. The user can tailor what types of notifications he or she receives, as well as which e-mail addressable device they are sent to. Most users send the alerts to their cell phones. However, one user who was traveling to India for a 4-week trip chose to send the alerts to his Internet e-mail account. Based on the content of the alert, he could decide whether to dial into the firewall protected Lotus Notes e-mail system to read the entire message. Reminders (also referred to as *to-dos*) are the most frequently used type of reminder. Calendar alerts, which consist of notifications of meetings and conference calls are the second most common category of alert. Users can also opt to be notified when they receive an urgent e-mail message (the alert contains the sender and subject) or when a voice mail message is received. The last category of alerts consists of user-defined rules, referred to as custom rules in Figure 6 (e.g., notify of any message where sender = X). Figure 7 shows the number of voice mail messages delivered by month. This is a more recent feature of the system, which is why there are fewer months of data.

8.2. Design Changes

An analysis of our growing user population and of the usage logs revealed two important facts that drove a design change in the interaction style. The first is that we have a large percentage of new users as well as a fair number of infrequent users. Although some users have incorporated remote and mobile message access into their daily work routine, others call the system less frequently depending on their circumstances (e.g., when stuck in traffic or on a Sunday night). Also, because this prototype is in the process of being rolled out within IBM Research, we are enrolling new users every week. Although we had initially expected the system to be ac-

cessed daily by all users, we have learned that users develop different usage patterns depending on lifestyles, length of commute, job responsibilities, and affinity for new technologies. The less frequent users were sometimes forgetting what they had learned about interacting with the system from previous calls, and new users needed guidance to find their way around the services that are available. Therefore, we added more structure to the dialog to guide users. We changed our opening system prompt after logon from, "I'm ready to help" to "Where shall we start: messages, calendar, greeting, or reminders." Although this latter prompt is reminiscent of less conversational dialogs, the user is not restricted to saying one of the four choices. The user can reply with unconstrained input (e.g., "I'd like to hear my first voice mail message"). If the user opts to follow the path set out before him, we continue with the more directed interaction for the first few turns. A quick examination of the call logs (a more complete study is in progress) shows that for about 35% of recent calls, users respond with one of the four choices (i.e., messages, calendar, greeting, or reminders). In about an equal number of calls, users use DTMF in response to the initial system prompt. In only about 28% of calls do users respond with a spoken utterance that is different than one of the four presented choices. These percentages are subject to further analysis.

9. FUTURE WORK

There are a number areas to explore with the conversational interface to increase robustness. We would like to extend the NLU models to support ambiguous and subtle requests. If individual word recognition scores were available, we could examine them to help identify subtle differences in phrasing (e.g., "send a message" vs. "send that message" or "new message" vs. "new messages"). Furthermore, it would be advantageous to use the NLU models to deal with ambiguous utterances such as, "new e-mail." If the recognizer could discern the tone of the request and include a period or a question mark, the ambiguity could be eliminated. Second, we would like to enhance the system to support compound requests such as, "Delete that and read the next message." Compound utterances are common ways of interacting and providing a nice shortcut mechanism.

The current prototype only parses user input for content. To develop a better system, we must also parse the data that is read to the users (e.g., messages, calendar entries) so that information contained in the messages can be incorporated into the interaction. For example, after listening to a meeting invitation message, the user should be able to say, "Put that on my calendar." We have just begun work in this area, but have not yet deployed anything with our users. Our current voice mail system is simply a voice recording capture mechanism. We would like to see this application evolve into a fully conversational application, providing services that are similar to what a human assistant would do. Call screening, directory lookup, calendar access, and emergency hand-over to a human are just some of the features that we are considering. Finally, we would like to begin work on a multimodal interface, which would combine audio and visual modalities to deliver content and accept input from the user.

There are also several features that we would like to add to the dialog to improve the overall usability of the interaction. The first involves filling the gap while the server is processing the user request with a sound or music to indicate that the system is busy and not ready for user input. Currently, after the user speaks, there is a delay while the system analyzes the request and retrieves the requested information. While the system is working, there is silence (commonly referred to as latency), which is longer than the gaps in human–human conversations. The latency can occasionally cause the user to repeat his request, because silence in human–human interaction signals lack of understanding or not having heard. Even if the user does not repeat himself, the gap in the interaction can give the impression of system unresponsiveness. Another feature that we would like to add to the dialog is the ability to ask about messages based on the subject line of the message. Therefore, although the system currently supports queries such as, "Do I have any messages from Bob Jones?," it does not support requests along the lines of, "Are there any messages about today's meeting?"

Finally, there are several hard problems involved with parsing e-mail messages that will require additional work. These include recognizing signatures and skipping the contact and address information when reading the message, clearly presenting information contained in the note histories, and dealing with attachments. We would like to create an improved initial summarization of the inbox, and allow users to switch their configuration options (e.g., turn notification on or off) from their phone instead of their desktop.

10. CONCLUSION

Speaker-independent, large vocabulary recognition over a cell phone with varying degrees of cell coverage is at the cutting edge of what is possible given today's speech recognition engines. Dialog between two humans under these conditions is far from error free; however, over the course of the past several thousand years we have perfected many ways to detect misunderstandings and adjust for them in conversations. Natural dialog between a machine and a human is still far from resembling a human–human conversation, but it is making progress.

Like many solutions that use emerging technologies, we have certain users who quickly integrated this new modality into their daily work routine, and others who are still finding usage patterns that they are most comfortable with. Given the growing number of subscribers that are enrolling, it is clear that there is a need for users to have quick access to messages and to stay in touch with the current status of their e-mail and voice mail. Notification, concise presentation, and summarization are three key factors in delivering unified message content with synthetic speech. Although users appear to differ in their preferred means of interacting with the speech system (DTMF, conversational utterances, or directed dialog), a given user will interact with the system in a very consistent pattern from one call to the next.

E-mail is a text-intensive application that is challenging to present in an auditory fashion. We have developed a matrix of conversational functions to support the same type of browsing functions that we use when working with text visually.

There are many areas within the domain of speech-based pervasive computing applications that we are still exploring. These include an examination of the trade-offs between the use of DTMF and speech (when is DTMF used and why), intelligent parsing and presentation of complex messages with synthetic speech, and the correct balance between conversational and directed prompts. We hope to gain further knowledge of these areas and more, through the continued deployment of the MA application within IBM.

REFERENCES

Ballentine, B., & Morgan, D. (1999). *How to build a speech recognition application: A style guide for telephony dialogues.* San Ramon, CA: Enterprise Integration Group, Inc.

Clark, H. (1993). *Arenas of language use.* Chicago: University of Chicago Press.

Davies, K., Donovan, R., Epstein, M., Franz, M., Ittycheriah, A., & Jan, E. (1999). The IBM conversational telephony system for financial application. *Proceedings of Eurospeech '99, 1,* 275–278.

Dertouzos, M. (1999). The future of computing. *Scientific American, 281,* 52–55.

Esler, M., Hightower, J., Anderson, T., & Borriello, G. (1999). Next century challenges: data-centric networking for invisible computing; The Portolano Project at the University of Washington. *Conference Proceedings for Mobicom, 5th Annual ACMIIEEE,* 256–262.

Gong, L., & Lai, J. (2001). Shall we mix synthetic speech and human speech? Impact on users' performance, perception, and attitude. *Proceedings of the CHI 2001 Conference on Human Factors in Computing Systems, 3,* 158–166.

Kamm, K., & Helander, M. (1997). Design issues for interfaces using voice input. In M. Helander, T. Landauer, & P. V. Prabhu (Eds.), *Handbook of human–computer interaction* (pp. 1043–1060). Amsterdam: Elsevier-North Holland.

Norman, D. (2001). Cyborgs. *Communications of the ACM, 44,* 36–37.

Raman, B., Katz, R., & Joseph, A. (2000). Universal inbox: Providing extensible personal mobility and service mobility in an integrated communication network. *Third IEEE Workshop on Mobile Computing Systems and Applications* 95–106.

Reeves, B., & Nass, C. (1996). *The media equation: How people treat computers, television, and new media like real people and places.* New York: Cambridge University Press.

Schmandt, C. (1994). *Voice communication with computers: Conversational systems.* New York: Van Nostrand Reinhold.

Schmandt, C., Marmasse, N., Marti, S., Sawhney, N., & Wheeler, S. (2000). Everywhere messaging. *IBM Systems Journal, 39,* 660–677.

Seneff, S., Lau, R., & Polifroni, J. (1999). Organization, communication, and control in the GALAXY-II conversational system. *Proceedings Eurospeech '99, 3,* 1271–1274.

Shneiderman, B. (1998). *Designing the user interface. Strategies for effective human–computer interaction* (3rd ed.). Reading, MA: Addison-Wesley.

Wang, H., Raman, B., Chuah, C., Biswas, R., Gummadi, R., Hohlt, B., et al. (2000). ICEBERG: An Internet-core network architecture for integrated communications. *IEEE Personal Communications: Special Issue on IP-based Mobile Telecommunication Networks, 7,* 10–19.

Yankelovich, N. (1996). How do users know what to say? *ACM Interactions, 3,* 32–43.