

Charting Past, Present, and Future Research in Ubiquitous Computing

GREGORY D. ABOWD and ELIZABETH D. MYNATT

Georgia Institute of Technology

The proliferation of computing into the physical world promises more than the ubiquitous availability of computing infrastructure; it suggests new paradigms of interaction inspired by constant access to information and computational capabilities. For the past decade, application-driven research in ubiquitous computing (ubicomp) has pushed three interaction themes: *natural interfaces*, *context-aware applications*, and *automated capture and access*. To chart a course for future research in ubiquitous computing, we review the accomplishments of these efforts and point to remaining research challenges. Research in ubiquitous computing implicitly requires addressing some notion of scale, whether in the number and type of devices, the physical space of distributed computing, or the number of people using a system. We posit a new area of applications research, *everyday computing*, focussed on scaling interaction with respect to time. Just as pushing the availability of computing away from the traditional desktop fundamentally changes the relationship between humans and computers, providing *continuous* interaction moves computing from a localized tool to a constant companion. Designing for continuous interaction requires addressing interruption and resumption of interaction, representing passages of time and providing associative storage models. Inherent in all of these interaction themes are difficult issues in the *social implications* of ubiquitous computing and the challenges of *evaluating* ubiquitous computing research. Although cumulative experience points to lessons in privacy, security, visibility, and control, there are no simple guidelines for steering research efforts. Akin to any efforts involving new technologies, evaluation strategies form a spectrum from technology feasibility efforts to long-term use studies—but a user-centric perspective is always possible and necessary.

Categories and Subject Descriptors: H.5.2 [Information Interfaces and Presentation]: User Interfaces—*Evaluation/methodology; Interaction styles; Prototyping*; H.5.m [Information Interfaces and Presentation]: Miscellaneous; J.m [Computer Applications]: Miscellaneous; K.4.2 [Computers and Society]: Social Issues

General Terms: Human Factors

Additional Key Words and Phrases: Augmented reality, context-aware applications, capture and access, evaluation, everyday computing, natural interfaces, social implications, ubiquitous computing, user interfaces

Authors' address: College of Computing & GVU Center, Georgia Institute of Technology, Atlanta, GA 30332-0280; email: abowd@cc.gatech.edu; mynatt@cc.gatech.edu.

Permission to make digital/hard copy of part or all of this work for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial advantage, the copyright notice, the title of the publication, and its date appear, and notice is given that copying is by permission of the ACM, Inc. To copy otherwise, to republish, to post on servers, or to redistribute to lists, requires prior specific permission and/or a fee.

© 2000 ACM 1073-0516/00/0300-0029 \$5.00

1. INTRODUCTION

Weiser introduced the area of ubiquitous computing (ubicomputing) and put forth a vision of people and environments augmented with computational resources that provide information and services when and where desired [Weiser 1991]. For the past decade, ubicomputing researchers have attempted this augmentation with the implicit goal of assisting everyday life and not overwhelming it. Weiser's vision described a proliferation of devices at varying scales, ranging in size from hand-held "inch-scale" personal devices to "yard-scale" shared devices. This proliferation of devices has indeed occurred, with commonly used devices such as hand-held personal digital assistants (PDAs), digital tablets, laptops, and wall-sized electronic whiteboards. The development and deployment of necessary infrastructure to support continuous mobile computation is arriving.

Another aspect of Weiser's vision was that new applications would emerge that leverage off these devices and infrastructure. Indeed, ubicomputing promises more than just infrastructure, suggesting new paradigms of interaction inspired by widespread access to information and computational capabilities. In this article, we explore how this applications perspective has evolved in the decade since the start of the Ubiquitous Computing project at Xerox PARC. Specifically, we review the accomplishments and outline remaining challenges for three themes:

- We desire *natural interfaces* that facilitate a richer variety of communications capabilities between humans and computation. It is the goal of these natural interfaces to support common forms of human expression and leverage more of our implicit actions in the world. Previous efforts have focused on speech input and pen input, but these interfaces still do not robustly handle the errors that naturally occur with these systems; also these interfaces are too difficult to build.
- Ubicomp applications need to be *context-aware*, adapting their behavior based on information sensed from the physical and computational environment. Many applications have leveraged simple context, primarily location and identity, but numerous challenges remain in creating reusable representations of context, and in creating more complex context from sensor fusion and activity recognition.
- Finally, a large number of ubicomputing applications strive to automate the *capture* of live experiences and provide flexible and universal *access* to those experiences later on.

Undertaking issues of scale is implicit in the definition of ubicomputing research. Weiser defined the notion of scale as a broad space of *computational devices* [Weiser 1991]. Likewise, scaling systems with respect to distribution of computation into *physical space* reinforces the desire to break the human away from desktop-bound interaction. Requirements for critical-mass acceptance and collaboration imply scaling with respect to *people*. A final dimension, *time*, presents new challenges for scaling a

system. Pushing the availability of interaction to a “24-by-7” (24 hours a day, 7 days a week) basis uncovers another class of largely unexplored interactions that will also push ubicomp research into the next century. To address scaling with respect to time, in Section 5, we introduce a new theme, called *everyday computing*, that promotes informal and unstructured activities typical of much of our everyday lives. These activities are continuous in time, a constant ebb and flow of action that has no clear starting or ending point. Familiar examples are orchestrating tasks, communicating with family and friends, and managing information.

The structure of this article follows the evolutionary path of past work in ubicomp. The first step in this evolution, demonstrated by the PARCTab [Want et al. 1995] and Liveboard [Elrod et al. 1992], is computers encased in novel form factors. Often these computational appliances push on traditional areas in computer science such as networking and operating systems. Since these new form factors often do not work well with traditional input devices such as the keyboard and mouse, developing new, and more natural, input capabilities is the next step. An example of this work is the pen-based shorthand language Unistroke for the PARCTab [Goldberg and Richardson 1993]. After some initial demonstrations, infrastructure is needed to deploy these devices for general use. For example, numerous tour guide systems that mimic the first use of Active Badges [Want et al. 1992] have been built and deployed for real use.

It is at this point that application designers begin working with these new systems to develop novel uses, often focusing on implicit user input to minimize the intrusion of technology into everyday life. The objective of this application-centered research is to understand how everyday tasks can be better supported, and how they are altered by the introduction of ubiquitous technologies. For example, ubicomp applications in support of common meeting tasks at PARC (through the Tivoli project) have resulted in new ways to scribe and organize materials during meetings. Capture environments in educational settings have provided more opportunities to understand the patterns of longer-term reviewing tasks over large multimedia records. Applications of wearable computers initially emphasized constant access to traditional individual tasks, such as accessing email. More recent applications have attempted to augment an individual’s memory and provide implicit information sharing between groups. The direction of applications research, what Weiser himself deemed the ultimate purpose for ubicomp research, is deeply influenced by authentic and extended use of ubicomp systems.

Today we are just starting to understand the implications of continuous immersion in computation. The future will hold much more than constant availability of tools to assist with traditional, computer-based tasks. Whether we wear computers on our body, or have them embedded in our environment, the ability of computers to alter our perception of the physical world, to support constant connectivity to distant people and places, to provide information at our fingertips, and to continuously partner with us

in our thoughts and actions offers much more than a new “killer app”—it offers the possibility of a killer existence.

Overview. In this article, we investigate the brief history of ubiquitous computing through exploration of the above-mentioned interaction themes—natural interfaces, context-aware computing, and automated capture and access for live experiences. In addition to reviewing the research accomplishments in these application research themes, we also outline some of the remaining research challenges for HCI researchers to pursue in the new millennium. We then explain the necessity for ubicomp research to explore continuous everyday activities. This area of research motivates applications that build off of the three earlier themes and moves ubicomp more into the realm of everyday computing characterized by continuously present, integrative, and unobtrusive interaction. Inherent in all of these interaction themes are difficult issues in the social implications of ubiquitous computing and the challenges of evaluating ubiquitous computing research. We conclude with our reflections on these issues and description, via case studies, of our current strategies for evaluation of ubicomp systems.

2. COMPUTING WITH NATURAL INTERFACES

Ubiquitous computing inspires application development that is “off the desktop.” Implicit in this mantra is the assumption that physical interaction between humans and computation will be less like the current desktop keyboard/mouse/display paradigm and more like the way humans interact with the physical world. Humans speak, gesture, and use writing utensils to communicate with other humans and alter physical artifacts. These natural actions can and should be used as explicit or implicit input to ubicomp systems.

Computer interfaces that support more natural human forms of communication (e.g., handwriting, speech, and gestures) are beginning to supplement or replace elements of the GUI interaction paradigm. These interfaces are lauded for their learnability and general ease of use, and their ability to support tasks such as authoring and drawing without drastically changing the structure of those tasks. Additionally, they can be used by people with disabilities for whom the traditional mouse and keyboard are less accessible.

There has been work for many years in speech-related interfaces, and the emerging area of perceptual interfaces is being driven by a long-standing research community in computer vision and computational perception [Turk 1997; 1998]. Pen-based or free-form interaction is also realizing a resurgence after the failure of the first generation of pen computing. More recently, researchers have suggested techniques for using objects in the physical world to manipulate electronic artifacts, creating so-called graspable [Fitzmaurice et al. 1995] or tangible user interfaces [Ishii and Ullmer 1997]. Harrison et al. [1998] have attached sensors to computational devices in order to provide ways for physical manipulations of those devices

to be interpreted appropriately by the applications running on those devices. Applications that support natural interfaces will leverage off of all of these input and output modalities. Instead of attempting to review the impressive amount of work in natural interfaces, we focus on two issues that are important for enabling the rapid development of effective natural interfaces. One important area we will not discuss is that of multimodal integration, a theme with its own conferences and journals already.

2.1 First-Class Natural Data Types

To ease the development of more applications with natural interfaces, we must be able to handle other forms of input as easily as keyboard and mouse input. The raw data or signals that underlie these natural interfaces—audio, video, ink, and sensor input—need to become first-class types in interactive system development. As programmers, we expect that any user interface toolkit for development provides a basic level of support for “fundamental” operations for textual manipulation, and primitives for keyboard and mouse interaction. Similarly, we need basic support for manipulating speech—such as providing speaker pause cues or selection of speech segments or speaker identification—as well as for video and ink and other signals, such as physical device manipulations detected by sensors.

Take, for example, freeform, pen-based interaction. Much of the interest in pen-based computing has focussed on recognition techniques to convert the “ink” from pen input to text. However, some applications, such as personal note-taking, do not require conversion from ink to text. In fact, it can be intrusive to the user to convert handwriting into some other form. Relatively little effort has been put into standardizing support for freeform, pen input. Some formats for exchanging pen input between platforms exist, but little effort has gone into defining effective mechanisms for manipulating the freeform ink data type within programs.

What kinds of operations should be supported for a natural data type such as ink? The Tivoli system provided basic support for creating ink data and distinguishing between uninterpreted, freeform ink data and special, implicitly structured gestures [Minneman et al. 1995; Moran et al. 1995; 1996; 1997a]. Another particularly useful feature of freeform ink is the ability to merge independent strokes together as they form letters, words, and other segments of language. In producing Web-based notes in Classroom 2000 (discussed in more detail below), for example, we wanted annotations written with a pen by a lecturer to link to the audio or video of what was said or seen at that same time during a lecture [Abowd 1999]. The annotations are timestamped, but it is not all that useful to associate an individual penstroke to the exact time it was written in class. We used a temporal and spatial heuristic to statically merge penstrokes together and assign them a more meaningful, word-level timestamp [Abowd et al. 1998b]. Chiu and Wilcox [1998] have produced a more general and dynamic algorithm, based on hierarchical agglomeration, to selectively link audio and ink. These structuring techniques need to become standard and avail-

able to all applications developers who wish to create freeform, pen-based interfaces. And as the work of Chiu and Wilcox demonstrates, some of the structuring techniques can apply to more than one natural data type. We must also think about primitive operations that combine different natural data types.

2.2 Error-Prone Interaction for Recognition-Based Interaction

When used for recognition-based tasks, natural interfaces come with a new set of problems: they permit new and more kinds of mistakes. When recognition errors occur, the initial reaction of system designers is to try to eliminate them, e.g., by improving recognition accuracy. However, Van Buskirk and LaLomia [1995] found that a reduction of 5–10% in the absolute error rate is necessary before the majority of people will even notice a difference in a speech recognition system.

Worse yet, eliminating errors may not be possible. Even humans make mistakes when dealing with these same forms of communication. For an example, consider handwriting recognition. Even the most expert handwriting recognizers (humans) can have a recognition accuracy as low as 54% [Schomaker 1994]. Human accuracy increases to 88% for cursive handwriting [Schomaker 1994], and 96.8% for printed handwriting [Frankish et al. 1992], but it is never perfect. This evidence all suggests that computer handwriting recognition will never be perfect. Indeed, computer-based recognizers are even more error-prone than humans. The data they can use is often less fine-grained than what humans are able to sense. They have less processing power. And variables such as fatigue can cause usage data to differ significantly from training data, causing reduced recognition accuracy over time [Frankish et al. 1992].

On the other hand, recognition accuracy is not the only determinant of user satisfaction. Both the complexity of error recovery dialogues [Zajicek and Hewitt 1990] and the value-added benefit for any given effort [Frankish et al. 1995] affect user satisfaction. For example, Frankish et al. [1995] found that users were less frustrated by recognition errors when the task was to enter a command in a form than when they were writing journal entries. They suggest that the pay-back for entering a single word in the case of a command is much larger when compared with the effort of entering the word in a paragraph of a journal entry.

Error handling is not a new problem. In fact, it is endemic to the design of computer systems that attempt to mimic human abilities. Research in the area of error handling for recognition technologies must assume that errors will occur, and then answer questions about the best ways to deal with them. Several research areas for error handling of recognition-based interfaces have emerged:

—*Error reduction*: This involves research into improving recognition technology in order to eliminate or reduce errors. It has been the focus of extensive research, and could easily be the subject of a whole paper on its own. Evidence suggests that its holy grail, the elimination of errors, is probably not achievable.

- Error discovery*: Before either the system or the user can take any action related to a given error, one of them has to know that the error has occurred. The system may be told of an error through explicit user input, and can help the user to find errors through effective output of uncertain interpretations of recognized input. Three techniques are used to automate such error discovery—thresholding of confidence measures, historical statistics [Marx and Schmandt 1994], and explicit rule specification [Baber and Hone 1993].
- Reusable infrastructure for error correction*: Toolkits provide reusable components and are most useful when a class of common, similar problems exists. Interfaces for error handling would benefit tremendously from a toolkit that presents a library of error-handling techniques of recognition-based input. Such a toolkit would have to handle the inherent ambiguities that arise when multiple interpretations are generated for some raw input. A prototype toolkit has been proposed by Mankoff et al. [2000] to support reusable recovery techniques, but many challenges remain.

3. CONTEXT-AWARE COMPUTING

Two compelling early demonstrations of ubicomp were the Olivetti Research Lab's Active Badge [Want et al. 1992] and the Xerox PARCTab [Want et al. 1995], both location-aware appliances. These devices leverage a simple piece of context, user location, and provide valuable services (automatic call forwarding for a phone system, automatically updated maps of user locations in an office). Whereas the connection between computational devices and the physical world is not new—control systems and autonomously guided satellites and missiles are other examples—these simple location-aware appliances are perhaps the first demonstration of linking implicit human activity with computational services that serve to augment general human activity.

Location is a common piece of context used in application development. The most widespread applications have been GPS-based car navigation systems and handheld “tour guide” systems that vary the content displayed (video or audio) by a hand-held unit given the user's physical location in an exhibit area [Abowd et al. 1997; Bederson 1995; Cheverst et al. 1998; Opperman and Specht 1998]. Another important piece of context is recognizing individual objects. Earlier systems focused on recognizing some sort of barcode or identifying tag, while recent work includes the use of vision-based recognition. Fitzmaurice et al. [1993; 1995] demonstrated using a hand-held device to “see inside” walls and pieces of machinery. Rekimoto and Nagao's [1995] NaviCam (see Figure 1) recognized color barcodes overlaying additional information about objects on a hand-held video display. Recent efforts [Jebara et al. 1997] are attempting to substitute visual object recognition strategies so that objects do not have to be individually tagged.



Fig. 1. In Rekimoto and Naga's [1995] NaviCam system, a handheld device recognizes tagged objects and then overlays context-sensitive information.

Although numerous systems that leverage a person's identity and/or location have been demonstrated, these systems are still difficult to implement. Salber et al. [1999] created a "context toolkit" that simplifies designing, implementing, and evolving context-aware applications. This work emphasizes the strict separation of context sensing and storage from application-specific reaction to contextual information, and this separation facilitates the construction of context-aware applications. Mynatt et al. [1998] point to the common design challenge of creating a believable experience with context-aware interfaces noting that the responsiveness of the interface is key to the person associating additional displays with their movements in the physical world.

In many ways, we have just scratched the surface of context-aware computing with many issues still to be addressed. Here we will discuss challenges in incorporating more context information, representing context, ubiquitous access to context sensing and context fusion, and the coupling of context and natural interaction to provide effective augmented reality.

3.1 What Is Context?

There is more to context than position and identity. Most context-aware systems still do not incorporate knowledge about time, history (recent or

long past), other people than the user, as well as many other pieces of information often available in our environment. Although a complete definition of context is illusive, the “five W’s” of context are a good minimal set of necessary context:

- Who*: Current systems focus their interaction on the identity of one particular user, rarely incorporating identity information about other people in the environment. As human beings, we tailor our activities and recall events from the past based on the presence of other people.
- What*: The interaction in current systems either assumes what the user is doing or leaves the question open. Perceiving and interpreting human activity is a difficult problem. Nevertheless, interaction with continuously worn, context-driven devices will likely need to incorporate interpretations of human activity to be able to provide useful information.
- Where*: In many ways, the “where” component of context has been explored more than the others. Of particular interest is coupling notions of “where” with other contextual information, such as “when.” Some tour guide systems have theorized about learning from a history of movements in the physical world, perhaps to tailor information display based on the perceived path of interest by the user. Again these ideas need fuller exploration.
- When*: With the exception of using time as an index into a captured record or summarizing how long a person has been at a particular location, most context-driven applications are unaware of the passage of time. Of particular interest is understanding relative changes in time as an aid for interpreting human activity. For example, brief visits at an exhibit could be indicative of a general lack of interest. Additionally, when a baseline of behavior can be established, action that violates a perceived pattern would be of particular interest. For example, a context-aware home might notice when an elderly person deviated from a typically active morning routine.
- Why*: Even more challenging than perceiving “what” a person is doing is understanding “why” that person is doing it. Sensing other forms of contextual information that could give an indication of a person’s affective state [Picard 1997], such as body temperature, heart rate, and galvanic skin response, may be a useful place to start.

3.2 Representations of Context

Related to the definition of context is the question of how to represent context. Without good representations for context, applications developers are left to develop ad hoc and limited schemes for storing and manipulating this key information. The evolution of more sophisticated representations will enable a wider range of capabilities and a true separation of sensing context from the programmable reaction to that context.

3.3 The Ubiquity of Context Sensing—Context Fusion

An obvious challenge of context-aware computing is making it truly ubiquitous. Having certain context, in particular positioning information, has been shown useful. However, there are few truly ubiquitous, single-source context services. Positioning is a good example. GPS does not work indoors and is even suspect in some urban regions as well. There are a variety of indoor positioning schemes as well, with differing characteristics in terms of cost, range, granularity, and requirements for tagging, and no single solution is likely to ever meet all requirements.

The solution for obtaining ubiquitous context is to assemble context information from a combination of related context services. Such *context fusion* is similar in intent to the related, and well-researched, area of sensor fusion. Context fusion must handle the seamless handing off of sensing responsibility between boundaries of different context services. Negotiation and resolution strategies need to integrate information from competing context services when the same piece of context is concurrently provided by more than one service. This fusion is also required because sensing technologies are not 100% reliable or deterministic. Combining measures from multiple sources could increase the confidence value for a particular interpretation. In short, context fusion assists in providing reliable ubiquitous context by combining services in parallel, to offset noise in the signal, and sequentially to provide greater coverage.

3.4 Coupling Context-Aware and Natural Interaction—Augmented Reality

The goal of many context-aware applications is to allow the user to receive, in real-time, information based on actions in the physical world. The tour guide systems are a good example—the user’s movements in an exhibit triggered the display of additional, context-sensitive information. These applications typically used separate, albeit portable, devices that require attention away from the rest of the physical world. The best metaphor to describe these interactions is that the user is “probing the world with a tool,” similar to tools such as electronic stud finders and geiger counters.

By incorporating augmented vision and augmented hearing displays, as well as natural input such as voice and gesture, we will more closely integrate context-aware interaction with the physical world in which it resides [MacIntyre and Feiner 1996; MacIntyre and Mynatt 1998; Starner et al. 1997]. In these interactions, the system is modifying how a user perceives the physical world. This tighter integration of information and perception should allow for more natural, seamless, hands-busy, and serendipitous interaction (see Figure 2).

4. AUTOMATED CAPTURE AND ACCESS TO LIVE EXPERIENCES

Much of our life in business and academia is spent listening to and recording, more or less accurately, the events that surround us, and then trying to remember the important pieces of information from those events. There is clear value, and potential danger, in using computational re-

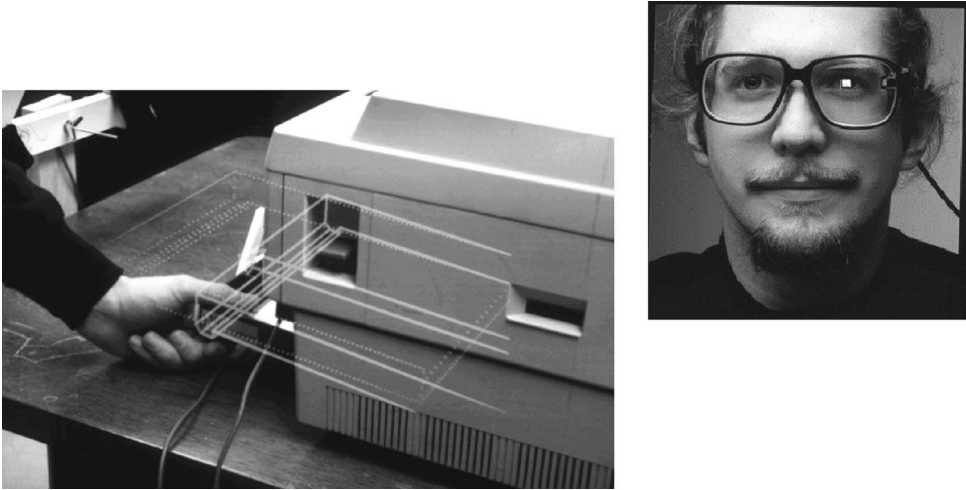


Fig. 2. In the KARMA system (on the left) augmented views required heavy, clunky, headmounted displays [Feiner et al. 1993]. Now lightweight glasses, such as the ones shown on the right above from MicroOptical, provide similar display capabilities.

sources to augment the inefficiency of human record-taking, especially when there are multiple streams of related information that are virtually impossible to capture as a whole manually. Tools to support automated capture of and access to live experiences can remove the burden of doing something humans are not good at (i.e., recording) so that they can focus attention on activities they are good at (i.e., indicating relationships, summarizing, and interpreting).

There has been a good deal of research related to this general capture-and-access theme, particularly for meeting-room/classroom environments and personal note-taking. Early work by Schmandt and Arons [1985] and Hindus and Schmandt [1992] captured audio from phone conversations and provided ways to access the content of the recorded conversations. The two systems, PhoneSlave and Xcapture, treated audio as uninterpreted data and were successful using simple techniques to provide informative overviews of live conversations. More recent research efforts have tried to capture other types of input, such as freeform ink. The Tivoli system used a suite of software tools to support a scribe at a meeting [Minneman et al. 1995; Moran et al. 1996; 1997b] as well as some electronic whiteboard technology—the LiveBoard [Elrod et al. 1992]—to support group discussion. Artifacts produced on the electronic whiteboard during the meeting are timestamped. This temporal information is used after the meeting to index into recorded audio or video, thus providing the scribe a richer set of notes from the meeting. Similar integration between recorded ink annotations and audio/video is supported in Classroom 2000 for university lectures [Abowd 1999; Abowd et al. 1998a; 1998b], with a greater emphasis on automating the postproduction of captured material into universally accessible interfaces for a large population of students. Other capture systems, such as Authoring on the Fly [Bacher and Ottmann 1996] and Cornell’s

Lecture Browser [Mukhopadhyay and Smith 1999], also focus on capture of presentations with attention to capturing arbitrary program interactions and production-quality video capture from multiple sources.

These systems focus on the capture of a public, group experience. Other capture systems, such as Marquee [Weber and Poon 1994], Filochat [Whittaker et al. 1994], We-Met [Wolf and Rhyne 1992], the Audio Notebook [Stifelman 1996; 1997], Dynamite [Wilcox et al. 1997], NotePals [Davis et al. 1999], and MRAS [White et al. 1998], focus on capture for the individual. StuPad [Truong et al. 1999] was the first system to provide mixed public and personal capture.

Most of the above efforts produce some sort of multimedia interface to review the captured experience. By focusing on this postproduction phase, some systems provide automated support for multiple camera fusion, integration of various presentation media, and content-based retrieval mechanisms to help search through a large repository of captured information. The postproduction results can then be accessed through a multimedia interface, typically distributed via the Web. Abowd [1999] provides a review of some of these research and commercial systems.

In all of these cases, the emphasis on ubiquity is clearly seen in the separate capture and access phases. Electronic capture is moved away from traditional devices, like the keyboard, and brought closer to the user in the form of pen-based interfaces or actual pen and paper. Input in the form of voice and gesture is also accepted and is either treated as raw data or further interpreted to provide more understanding of the captured experience.

4.1 Challenges in Capture and Access

Despite substantial research and advances in automated capture systems, there are a number of open research issues, that we summarize here. We separate out issues primarily associated with capture from those primarily associated with access.

4.1.1 Capture. We have mentioned earlier the importance of having a good driving application for ubicomp research. In the capture domain, the main compelling applications have been for meeting support and education/training. These are indeed compelling application areas. In particular, our evidence in Classroom 2000 points to overwhelming acceptance of capture from the student perspective [Abowd 1999]. There are many more possibilities, however, for exploring capture in equally compelling domains:

—Many of us record the special events in our lives—vacations, birthday parties, visits from relatives and friends—and we often spend time, years later, reflecting and remembering the events through the recordings on film and in diaries. How many times have we wished we had a camera at a particularly precious moment in our lives (a child's first steps) only to fumble for the recording device and miss the moment? How difficult is it sometimes to find the picture or film of a significant event?

- In many collaborative design activities, the critical insights or decisions are often made in informal settings and are usually not documented properly. Technical exchanges often flow quite freely in opportunistic encounters. Even in more formal design meetings, the rich exchange of information and discussions around artifacts, such as storyboards or architectural recommendations, is often very poorly captured. Recently, we have begun experimenting with support to capture both informal brainstorming activities [Brotherton et al. 1999] and structured design meetings [Richter et al. 1999].
- Maintenance of a building might be better supported if we captured a record of the actual construction of the building—in contrast to the building plans. When repairs are needed, the appropriate technician could “replay” the construction and maintenance history of the relevant building artifact in order to determine the right course of repair.

With the exception of the Audio Notebook, NotePals and Cornell’s Lecture Browser, there has been little work on capturing artifacts in the physical world and making them easily accessible in the access phase. The emergence of low-cost capture hardware, such as the CrossPad™ and the mimio™ from Virtual Ink, will lead more researchers to work in this area.

Much of the capture currently being done is for what we would call *raw* streams of information that are captured mainly for the purpose of direct playback. No further analysis on those streams is done. However, it is often useful to *derive* additional information from a simple stream to provide a greater understanding of the live event. For example, Stifelman used results from discourse analysis to further segment the captured audio stream and make better prediction about when new topics commenced in a discussion [Stifelman 1997]. Similarly, Chiu and Wilcox [1998] proposed a hierarchical agglomeration technique for using pause detection to segment and associate both ink and audio. Other computational perception techniques can be used to analyze the simple audio, ink, or video signals.

Another application of signal analysis is to improve the recording of raw streams. How can we automate certain well-known production practices that merge multiple camera feeds into a single, coherent, high-quality video that can be viewed later? Single, fixed camera angles are not sufficient to capture the salient parts of a live experience, but when we scale a system like Classroom 2000 to an entire campus, we cannot afford to pay technicians to sit in each of the classrooms. The single biggest challenge here is being able to determine the focus of attention for the group, and more difficult, for each individual at a live event.

4.1.2 *Access*. In the access phase, we need to provide a number of playback capabilities. The simplest is to playback in real time, but there are often situations in which this is inappropriate or overly inefficient. In reviewing a lecture for an exam, a student does not always want to sit through an entire lecture again, but he or she might want to pinpoint a particular topic of discussion and replay only that portion. Alternatively, a

summarization of the experience which gleans salient points from across an entire captured session might be more appropriate.

Synchronization of multiple captured streams during playback is vital. Commercial streaming products, such as RealNetworks G2/SMIL[™] and Microsoft's MediaPlayer/ASF[™], are emerging standards to allow for powerful synchronization of programmer-defined media streams. However, it is not clear that any of these products will support the foreshadowing of streams so that a user can see what lies ahead in reviewing a stream. Such foreshadowing can help a user skim more quickly to a point of interest.

In most of the systems, the captured material is static upon reaching the access phase. Of course, there are often cases where annotating or revising captured material is appropriate, as well as then revising revised notes and so on. Although versioning is not a new problem to computer scientists, there are numerous challenges in providing an intuitive interface to multiple versions of captured material, especially when some of the material is already time-based such as audio and video. A timeline is an effective interface for manipulating and browsing a captured session, but when the time associated with a captured artifact is split up into a number of non-contiguous time segments, the usefulness of the timeline is at least questionable. Newer time-based interaction techniques, such as Lifestreams [Fertig et al. 1996], Timewarp [Edwards and Mynatt 1997], and time-machine computing [Rekimoto 1999] are good starting points.

Finally, and perhaps most challenging, as these systems move from personalized systems to capturing events in more public settings, privacy concerns for the capture and later access of this material increase. Although these issues must be addressed in the specific design of each system, we still need general techniques for tagging material and authenticating access. We will discuss these issues later in this article.

5. TOWARD EVERYDAY COMPUTING

Earlier, we described an emerging area of interaction research, *everyday computing*, which results from considering the consequences of scaling ubiquitous computing with respect to time. Just as pushing the availability of computing away from the traditional desktop fundamentally changes the relationship between humans and computers, providing *continuous* interaction moves computing from a localized tool to a constant presence. Our motivations for everyday computing stem from wanting to support the informal and unstructured activities typical of much of our everyday lives. These activities are continuous in time, a constant ebb and flow of action that has no clear starting or ending point. Familiar examples are orchestrating tasks, communicating with family and friends, and managing information.

Designing for everyday computing requires addressing these features of informal, daily activities:

—*They rarely have a clear beginning or end:* Either as a fundamental activity, such as communication, or as a long-term endeavor, such as

research in human-computer interaction, these activities have no point of closure. Information from the past is often recycled. Although new names may appear in an address book or new items on a to-do list, the basic activities of communication or information management do not cease. A basic tenet in HCI is designing for closure. Given a goal, such as spell-checking a document, the steps necessary to accomplish that goal should be intuitively ordered with the load on short-term memory held to a reasonable limit. The dialogue is constrained so that the goal is accomplished before the user begins the next endeavor. When designing for an activity, principles such as providing visibility of the current state, freedom in dialogue, and overall simplicity in features play a prominent role.

—*Interruption is expected*: Thinking of these activities as continuous, albeit possibly operating in the background, is a useful conceptualization. One side-effect is that resumption of an activity does not start at a consistent point, but is related to the state prior to interruption. Interaction must be modeled as a sequence of steps that will, at some point, be resumed and built upon. In addition to representing past interaction, the interface can remind the user of actions left uncompleted.

—*Multiple activities operate concurrently*: Since these activities are continuous, the need for context-shifting among multiple activities is assumed. Application interfaces can allow the user to monitor a background activity, assisting the user in knowing when he or she should resume that activity. Resumption may be opportunistic, based on the availability of other people, or on the recent arrival of needed information. For example, users may want to resume an activity based on the number of related events that have transpired, such as reading messages in a newsgroup only after a reasonable number of messages have been previously posted. To design for background awareness, interfaces should support multiple levels of “intrusiveness” in conveying monitoring information that matches the relative urgency and importance of events. Current desktop interfaces only provide a small beginning in addressing these issues with multiple windows in a desktop interface. With minimal screen real estate, users must manage opening, closing, and restacking the many windows associated with a variety of tasks. Simple awareness cues are included in some desktop icons, indicating that new email has been received for example, but there are few controls for creating levels of notification to meet different awareness needs. The Rooms interface presented a compelling interface for spatially organizing documents and applications in multiple persistent working spaces [Card et al. 1999; Henderson et al. 1986]. This standard has yet to be met by current commercial “task” bars for changing application focus. A useful extension to Rooms would be both to provide awareness of “background” rooms, and to assist the user in remembering past activity when returning to a room.

- Time is an important discriminator:* Time is a fundamental human measuring stick although it is rarely represented in computer interfaces. Whether the last conversation with a family member was last week or five minutes ago is relevant when interpreting an incoming call from that person. When searching for a paper on a desk, whether it was last seen yesterday or last month informs the search. There are numerous ways to incorporate time into human-computer interfaces [Edwards and Mynatt 1997; Fertig et al. 1996; Rekimoto 1999]. As we try to regain our working state, interfaces can represent past events contingent on the length of time (minutes, hours, days) since the last interaction. As applications interpret real-world events, such as deciding how to handle an incoming phone call or to react to the arrival at the local grocery store, they can utilize timing information to tailor their interaction.
- Associative models of information are needed:* Hierarchical models of information are a good match for well-defined tasks, while models of information for activities are principally associative, since information is often reused on multiple occasions, from multiple perspectives. For example, assume you have been saving email from colleagues, friends, and family for a long time. When dealing with current mail, you may attempt to organize it into a hierarchy of folders on various topics. Over time, this organization has likely changed, resulting in a morass of messages that can be searched with varying degrees of success. Likewise, interfaces for to-do lists are often failures given the difficulty in organizing items in well-defined lists. Associative and context-rich models of organization support activities by allowing the user to reacquire the information from numerous points of view. These views are inherent in the need to resume an activity in many ways, for many reasons. For example, users may want to retrieve information based on current context such as when someone enters their office or when they arrive at the grocery store. They may also remember information relative to other current information, e.g., a document last edited some weeks ago or the document that a colleague circulated about some similar topic.

As computing becomes more ubiquitously available, it is imperative that the tools offered reflect their role in longer-term activities. Although principles in everyday computing can be applied to desktop interfaces, these design challenges are most relevant given a continuously changing user context. In mobile scenarios, users shift between activities while the computing resources available to them also vary for different environments. Even in an office setting, various tools and objects play multiple roles for different activities. For example, use of a computer-augmented whiteboard varies based on contextual information such as people present. Different physical objects such as a paper file or an ambient display can provide entry points and background information for activities. This distribution of interaction in the physical world is implicit in the notion of everyday computing, and thus clearly relevant to research in ubiquitous computing.

5.1 Synergy Among Themes

Research in everyday computing also continues to explore the three earlier interaction themes, but with the focus of designing a continuously available environment. Ishii's work in tangible media explores using natural interfaces to support communication and background awareness [Ishii and Ullmer 1997]. Current efforts in "Roomware" [Streitz et al. 1999] aim to create wall-sized and table-like interaction areas that support a greater range of informal human activity.

With respect to context-aware interaction, Audio Aura [Mynatt et al. 1998] is clearly related to previous tour guide systems, as a change in location triggers information delivery on a portable device. The motivation for Audio Aura, however, is to continuously augment the background auditory periphery of the user. By adding dynamic information about the activity of colleagues and communication channels (e.g., email), Audio Aura enhances the perceptible sphere of information available while the user continues with daily activities.

Likewise, applications for automated capture and access are moving into less structured environments. The Remembrance Agent [Rhodes 1997; Rhodes and Starner 1996] retrieves information based on physical context information including visual recognition. As the user can instruct the system about what to remember, the agent becomes a storehouse of everyday information that is continuously available, but indexed based on physical location. An unmet goal, first proposed by Bush [1945] is the design of personal memory containers that record continuously and later try to provide useful indices and summaries of the daily information they capture (see Lamming and Flynn [1994]).

5.2 Research Directions in Everyday Computing

Everyday computing offers many challenges to the HCI research community. In our current and future work, we are focusing on the following challenges:

—*Design a continuously present computer interface:* There are multiple models for how to portray computers that are ubiquitous, although none of these models are wholly satisfying. The notion of an *information appliance* [Norman 1998] typically reflects a special-purpose device that sits dumbly in the background without any knowledge of on-going activity. These interfaces often borrow from traditional GUI concepts and from consumer electronics. Computational systems that continue to operate in the background, perhaps learning from past activity and acting opportunistically, are typically represented as anthropomorphized *agents*. However it is doubtful that every interface should be based on dialogue with a talking head or human-oriented personality. Research in wearables explores *continually worn* interfaces [Starner et al. 1997], but these are limited by the current input and display technologies and are typically rudimentary text-based interfaces.

- Presenting information at different levels of the periphery of human attention:* Despite increasing interest in tangible media and peripheral awareness, especially in computer-supported collaborate work (CSCW) and wearable computing, current interfaces typically present a generic peripheral backdrop with no mechanism for the user, or the background task, to move the peripheral information into the foreground of attention. Our current design experiments are aimed at creating peripheral interfaces that can operate at different levels of the user's periphery.
- Connecting events in the physical and virtual worlds:* People operate in two disconnected spaces: the virtual space of email, documents, and Web pages and the physical space of face-to-face interactions, books, and paper files. Yet human activity is coordinated across these two spaces. Despite efforts as early as the Digital Desk [Wellner 1993] there is much work left to be done to understand how to combine information from these spaces to better match how people conceptualize their own endeavors.
- Modifying traditional HCI methods to support designing for informal, peripheral, and opportunistic behavior:* There is no one methodology for understanding the role of computers in our everyday lives. However, combining information from methods as different as laboratory experiments and ethnographic observations is far from simple. In our research and classroom projects, our goal is to learn by doing, by interrogating the results we derive from different evaluation strategies. We have consciously chosen a spectrum of methods that we believe match the questions we are asking. Learning how these methods inform each other and how their results can be combined will be an on-going effort throughout our work. We continue this discussion in the next section on evaluating ubicomp systems.

6. ADDITIONAL CHALLENGES FOR UBICOMP

Two important topics for ubicomp research—evaluation and social implications—cut across all themes of research, so we address them here.

6.1 Evaluating Ubicomp Systems

In order to understand the impact of ubiquitous computing on everyday life, we navigate a delicate balance between prediction of how novel technologies will serve a real human need and observation of authentic use and subsequent coevolution of human activities and novel technologies [Carroll and Rosson 1991]. Formative and summative evaluation of ubicomp systems is difficult for several reasons, which we will discuss. These challenges are why we see relatively little published from an evaluation or end-user perspective in the ubicomp community. A notable exception is the work published by Xerox PARC researchers on the use of the Tivoli capture system in the context of technical meetings [Moran et al. 1996]. Since research in ubiquitous computing will have limited impact in the HCI

community until it respects the need for evaluation, we have some advice for those wishing to undertake the challenges.

6.1.1 Finding a Human Need. The first major difficulty in evaluating a ubicomp system is simply having a reliable system to evaluate. The technology used to create ubicomp systems is often on the cutting edge and not well understood by developers, so it is difficult to create reliable and robust systems that support some activity on a continuous basis. Consequently, a good portion of reported ubicomp work remains at this level of unrobust demonstrational prototypes. This kind of research is often criticized as being technocentric, but as we will show, it is still possible to do good user-centered feasibility research with cutting-edge technology.

It is important in doing ubicomp research that a researcher build a compelling story, from the end-user's perspective, on how any system or infrastructure to be built will be used. The technology must serve a real or perceived human need, because, as Weiser [1993] noted, the whole purpose of ubicomp is to provide applications that serve the humans. The purpose of the compelling story is not simply to provide a demonstration vehicle for research results. It is to provide the basis for evaluating the impact of a system on the everyday life of its intended population. The best situation is to build the compelling story around activities that you are exposed to on a continuous basis. In this way, you can create a living laboratory for your work that continually motivates you to “support the story” and provides constant feedback that leads to better understanding of the use.

Designers of a system are not perfect, and mistakes will be made. Since it is already a difficult challenge to build robust ubicomp systems, you should not pay the price of building a sophisticated infrastructure only to find that it falls far short of addressing the goals set forth in the compelling story. You must do some sort of feasibility study of cutting-edge applications before sinking substantial effort into engineering a robust system that can be scrutinized with deeper evaluation. However, these feasibility evaluations must still be driven from an informed, user-centric perspective—the goal is to determine how a system is being used, what kinds of activities users are engaging in with the system, and whether the overall reactions are positive or negative. Answers to these questions will both inform future design as well as future evaluation plans. It is important to understand how a new system is used by its intended population before performing more quantitative studies on its impact.

Case Study: Xerox PARC's Flatland. Designing ubiquitous computing applications requires designers to project into the future how users will employ these new technologies. Although designing for a currently impossible interaction is not a new HCI problem, this issue is exacerbated by the implied paradigm shift in HCI resulting from the distribution of computing capabilities into the physical environment.

In our design work for Flatland [Mynatt et al. 1999], we employed ethnographic observations of whiteboard use in the office, coupled with questionnaires and interviews, to understand how people used their white-

boards on a daily basis (see Figure 3). The richness of the data from the observations was both inspirational in our design work and a useful constraint. For example, the notion of “hot spots,” portions of the board that users expect to change frequently, was the result of day-to-day observations of real whiteboard use. The data from the observations were key in grounding more in-depth user studies through questionnaires and interviews. Without these data, discussions would too easily slip into what users think they might do. By referring to two weeks of observational data, we were able to uncover and examine the details of daily practice.

Although the technology for our augmented whiteboard was not ready for deployment, or even user testing, we were able to gather a wealth of information from observations and interviews that critically informed our design.

Case Study: Audio Aura. The affordances and usability issues of novel input and output technologies are not well understood when they are first introduced. Often these technologies are still unusable for any real, long-term use setting. Nevertheless user-centric evaluations are needed to influence subsequent designs. In the design of Audio Aura [Mynatt et al. 1998], we were interested in exploring how peripheral awareness of relevant office activities could be enhanced through use of ambient sound in a mobile setting. Our combination of active badges, wireless headphones, and audio generation was too clunky for real adoption by long-term users. The headphones were socially prohibitive as they covered the ears with large, black shells. The capabilities for the development language, Java, to control sound presentation were too limited for creating rich auditory spaces. Nevertheless, we wanted to understand the potential interaction knowing that these technological limitations would be removed in the future.

We employed *scenarios* of interaction, based on informal observations of the Xerox PARC work environment, to guide our design and evaluation. These scenarios incorporated information about how people at PARC work together, including practices such as gathering at the coffee bistro, often dropping by people’s offices for impromptu conversations, and even the physical oddities of the building such as the long hallways that are the backbone of the layout. By grounding our scenarios in common practices, potential users could reflect on their daily activities when evaluating our designs. The scenarios also helped us understand a particular interaction issue: timing. In one of our scenarios, the communication path between the component technologies was not fast enough to meet the interaction demands. Although the speed could be increased, this modification required balancing a set of trade-offs, namely speed versus scalability, both important for our design goals. In short, the scenarios helped us understand the design space for further exploration.

6.1.2 Evaluating in the Context of Authentic Use. Deeper evaluation results require real use of a system, and this, in turn, requires a deployment into an authentic setting. The scaling dimensions that characterize ubicomp systems—device, space, people, or time—make it impossible to use

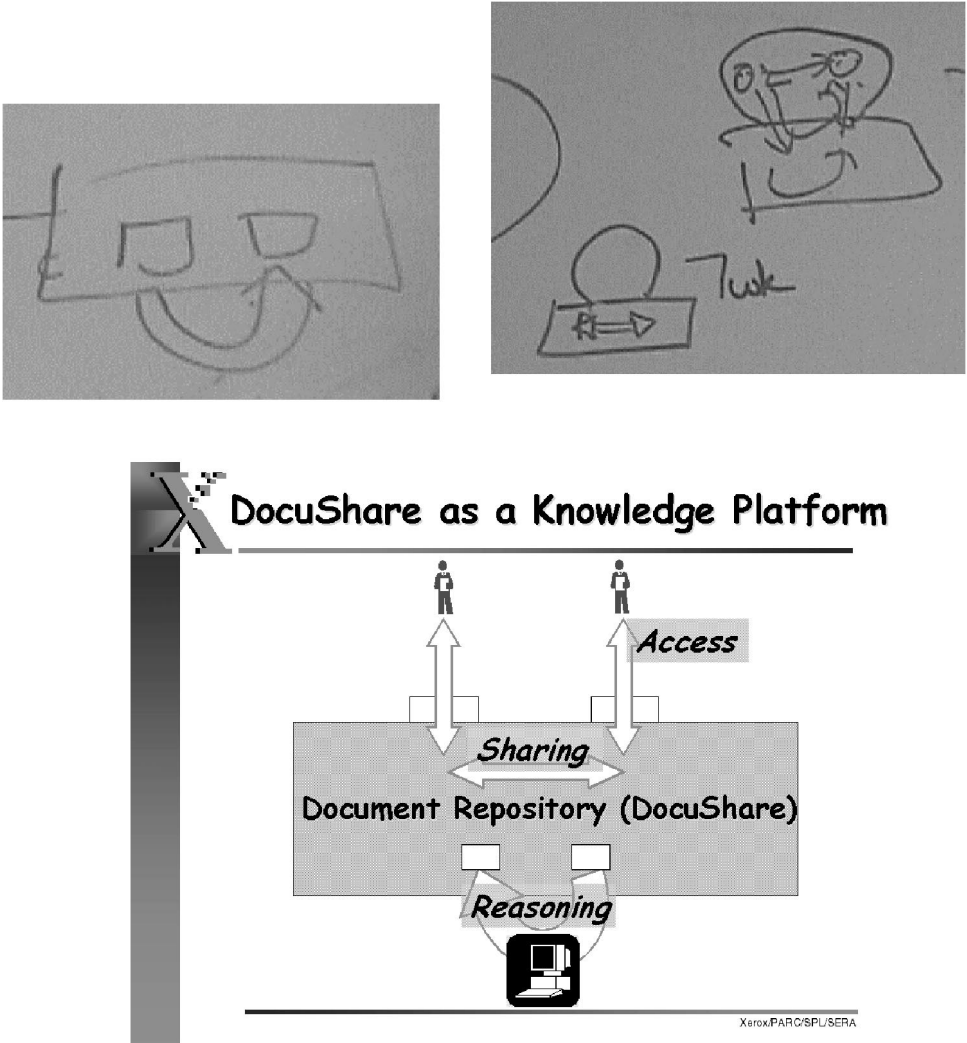


Fig. 3. In the design of Flatland, we used observations of whiteboard use to inform our design. Here whiteboard drawings on two different boards are used as the basis for discussing the complex concepts illustrated with more detail in the bottom slide.

traditional, contained usability laboratories. Effective evaluation, in which users are observed interacting with the system in routine ways, requires a realistic deployment into the environment of expected use.

Case Study: Classroom 2000. In this last case study, centered at Georgia Tech, we demonstrate a much longer term research project that evolved from early prototyping and feasibility studies into a more mature system that is currently used by a large population in a living classroom laboratory, shown in Figure 4. The project began in July 1995 with the intent of producing a system that would capture as much of the classroom experi-



Fig. 4. In the Classroom 2000 project, we have had the ability to learn from long-term actual use in the Georgia Tech educational environment.

ence as possible to facilitate later review by both students and teachers. In many lectures, students have their heads down, furiously writing down what they hear and see as a future reference. While some of this writing activity is useful as a processing cue for the student, we felt that it was desirable from the student and teacher perspective to afford the opportunity for students to lift their heads occasionally and engage in the lecture experience. The capture system was seen as a way to relieve some of the note-taking burden.

We needed to test the feasibility of this hypothesis quickly, so within six months of the project launch, we provided an environment to capture an entire course and observe whether our initial hypothesis was worth testing more vigorously. We learned some very valuable lessons during this first extended experience. The initial experiments included student note-taking devices that were clear distractions to the students [Abowd 1999], so we abandoned that part of the experiment, only to resume it in the past few months when the technology had caught up [Truong et al. 1999]. We also learned from this initial experience that in order to understand the impact of this capture system on teaching and learning we would have to gather usage data from a larger set of classes. This required significant engineering effort to create a robust and reliable capture system that by the Spring Quarter of 1997 was able to support multiple classes simultaneously. Today, after capturing over 100 courses with 30 different instructors, we have gained significant insight into how the system is used and what future directions to take [Abowd 1999]. As a direct result of these deeper evaluations, we know that the system encourages 60% of its users to modify their in-class note-taking behavior. We also know that not all of this

modified behavior is for the better. Taking no notes, for example, is not a good learning practice to reinforce. We know that it is time to reintroduce student note-taking units that can personalize the capture experience and also encourage better note-taking practices. We also know to facilitate more content-based retrieval and synchronized playback of the lecture experience. These insights have motivated further research efforts and established a long-term research project, eClass (see <http://www.cc.gatech.edu/fce/eclass>), that stands as a model for ubicomp research and automated capture and access.

6.1.3 Task-Centric Evaluation Techniques Are Inappropriate. Assuming an authentic deployment can occur, when users are comfortable with the service being provided and have developed habits for using the service, there is still the question of how to apply qualitative or quantitative evaluation methods. The majority of usability techniques are task-centric. If the user's tasks are known, then an evaluation is performed to determine the fitness of the system and its interface for completing that task. It is not at all clear how to apply task-centric evaluation techniques to informal everyday computing situations.

6.2 Social Issues for Ubiquitous Computing

We are pushing toward making it easier for computation to sense, understand, and react to phenomenon in the physical world and to record those phenomena. These enabling technologies carry with them numerous dangers, e.g., making it too easy for people to build systems that effectively spy on others without any controlling authority. Ubicomp researchers would be remiss if they undertook their work without understanding these issues. However, the fear of wrong-doing is not a call to cease all work in this area, but to work toward technological, design, and social solutions to address these concerns.

A basic concern about any information stored in a computer is knowing who can access and modify the contents. Where are the bits? Are they secure? Security and encryption schemes are part of the technological solutions available, especially as information is gleaned from the environment and transported over networks. Alternatively, work in wearable computing emphasizes a design approach—providing security by keeping the bits local (on the body) and removing the risks of transporting them over a public network.

One fear of users is the lack of knowledge of what some computing system is doing, or that something is being done “behind their backs.” Although the original vision of ubiquitous computing described computing as disappearing into the physical environment, this “invisibility” is counter to informing users about how they are being sensed. To assuage that fear, design solutions can be employed to make this information visible. For example, systems that sense physical phenomena and capture live situations should provide clear indicators that this sensing or recording is occurring. As these sensing and recording capabilities are more commonly

found, one challenge for everyday computing is to enable people to be aware of how they are being sensed. Just as people can ascertain their visibility in physical space (How public is this space? Are there windows?) we need cues to convey our visibility in virtual space.

The next step is to allow those being sensed or recorded to have control to either stop this activity or to at least control the distribution and use of the information. This challenge is related to the design of collaborative environments where the actions and roles of collaborators are fluid and difficult to articulate in a static snapshot. The capture, distribution, and use of information will be determined over time by the specific practices of people in workplace and home settings.

There are a number of reactions that system builders can have for handling the sensitive topic of when and what to capture. At Xerox PARC, one solution for capture was to agree to only capture the summary portions of technical meetings. In Classroom 2000, we defaulted to recording all of a lecture, but did not attempt to obtain good quality audio or video of anyone except the lecturer in the front of the room. In the Dynamite system from FX-PAL, the note-taker exercised control over which portions of the audio recording could be kept for future reference [Wilcox et al. 1997]. Though this last solution was first presented as a means of reducing the amount of storage requirements for high-fidelity audio, we see merit in this approach from an individual's perspective to enable an otherwise perfect capture system to "forget" some part of the past. An interesting challenge for collaborative situations is to figure out acceptable policies for "erasing" or "forgetting" some shared memory. A more positive slant on this issue would focus on ways to accommodate heightening awareness of particularly valuable segments of a captured experience in lieu of eliminating or forgetting parts of a captured history.

Although issues surrounding the appropriate use and dissemination of information are as old as the dawn of human communication, specific concerns stem from ubicomp making a new kind of information more generally available. The fact that computers can easily track our daily activities—a feat that previously required a large amount of human effort—is disconcerting at the least. In addition to addressing the above-mentioned concerns of security, visibility, and control, our approach is to create designed examples of appropriate and beneficial uses of this information. For example, one affordance of the low-quality video in early media spaces was that the amount of information conveyed was more socially appropriate. The not-real-time, grainy images met important needs for awareness and feelings of connectivity without violating privacy concerns. In the design of Audio Aura [Mynatt et al. 1998], we took great care in conveying *qualitative* information about the activities of colleagues. When stopping by someone's office, information that could be obtained by the system (e.g., this person has not been in his office for a few hours) was akin to the information that someone in a neighboring office could provide.

There are other social issues as well that are not as directly linked to privacy. For example, recording a meeting or a lecture can have both

positive and negative impact on those in attendance. On the positive side, knowledge of recording encourages people to be less reckless in their commentary. On the negative side, this same knowledge can cause people to refuse to contribute to a discussion for fear of saying something that would be regretted in the future. A more subtle problem was noticed in our extensive experience in Classroom 2000. Some students indicated that they chose not to ask questions in class because the answer was likely already discussed and it was up to the student to go back and listen to the lecture.

In general, social and legal practices continue to evolve in concert with technological and design innovations. In each situation people will compare the perceived benefits and costs of the uses of ubicomp technologies. For example, skiers and hikers choose to wear radio transponders so they can be located by rescue personnel. Firefighters benefit from understanding what each of them is doing and where they are located. Recent research details the calendaring practices at Sun Microsystems [Grudin and Palen 1997] where colleagues share extensive information about their daily collaborative activities. As discussed in the previous section on evaluation, our understanding of the social implications of these technologies will often come after people invent new, unforeseen, uses of these technologies. Although the sand is always shifting beneath us, attention to issues of security, visibility, control, and privacy should help ensure a more positive use of these technologies.

7. CONCLUDING THOUGHTS

In this article, we have attempted to outline the trajectory of ubicomp research in the decade since the inspiring work of Weiser and colleagues at Xerox PARC. We have identified three research themes for ubicomp, provided some background on significant achievements in those areas, as well as highlighted some of the remaining challenges. We have done this with the desire to motivate budding ubicomp researchers to attack some important and well-defined problems. We no doubt have left out some other important challenges for ubicomp research, and we look forward to seeing those problems articulated and solved by others.

Weiser [1993] claimed that the whole point of ubiquitous computing was to create compelling applications that would drive the development of devices and infrastructure. We agree in spirit with this claim, but want to promote a broader view that promotes the general-purpose utility (and challenge) of ubiquitous interaction with computational resources. The application or task-centric focus has been a fruitful one for HCI research. If we look at successful computing technology, however, it is not the case that a single application has driven critical-mass acceptance and deployment. What is the motivating application for the personal computer in our office or home, or for a Palm Pilot[™]? There are many applications, different for each person. The real goal for ubicomp is to provide many single-activity interactions that together promote a unified and continuous interaction between humans and computational services. The focus for the human at

any one time is not a single interface to accomplish some task. Rather, the interaction is more free-flowing and integrative, akin to our interaction with the rich physical world of people, places, and objects in our everyday lives.

ACKNOWLEDGMENTS

The authors would like to acknowledge the significant inspiration from the late Mark Weiser, who died somewhat suddenly during initial drafts of this article. Mark's vision and enthusiasm influenced a countless number of researchers, and we are indebted to his contributions in many ways. We would also like to acknowledge the interactions and influence of a number of colleagues from Xerox PARC and the Georgia Tech's Future Computing Environments Group within the College of Computing.

REFERENCES

- ABOWD, G. D. 1999. Classroom 2000: An experiment with the instruction of a living educational environment. *IBM Syst. J.* 38, 4, 508–530.
- ABOWD, G. D., ATKESON, C. G., HONG, J., LONG, S., KOOPER, R., AND PINKERTON, M. 1997. Cyberguide: a mobile context-aware tour guide. *Wireless Networks* 3, 5, 421–433.
- ABOWD, G. D., ATKESON, C. G., BROTHERTON, J., ENQVIST, T., GULLEY, P., AND LEMON, J. 1998a. Investigating the capture, integration and access problem of ubiquitous computing in an educational setting. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI '98, Los Angeles, CA, Apr. 18–23), M. E. Atwood, C.-M. Karat, A. Lund, J. Coutaz, and J. Karat, Eds. ACM Press/Addison-Wesley Publ. Co., New York, NY, 440–447.
- ABOWD, G. D., BROTHERTON, J., AND BHALODIA, J. 1998b. Automated capture, integration, and visualization of multiple media streams. In *Proceedings of the 1998 IEEE Conference on Multimedia and Computing Systems* (San Antonio, TX), IEEE Computer Society Press, Los Alamitos, CA, 54–63.
- BABER, C. AND HONE, K. S. 1993. Modeling error recovery and repair in automatic speech recognition. *Int. J. Man-Mach. Stud.* 39, 3 (Sept. 1993), 495–515.
- BACHER, C. H. R. AND OTTMANN, T. H. 1996. Tools and services for authoring on the fly. In *Proceedings of the Conference on Educational Multimedia and Hypermedia* (ED-MEDIA '96, Boston, MA, June 7–12),
- BEDERSON, B. B. 1995. Audio augmented reality: A prototype automated tour guide. In *Conference Companion for the ACM Conference on Human Factors in Computing Systems* (CHI '95, Denver, CO, May 7–11), J. Miller, I. Katz, R. Mack, and L. Marks, Eds. ACM Press, New York, NY, 210–211.
- BROTHERTON, J. A., ABOWD, G., AND TRUONG, K. N. 1999. Supporting capture and access interfaces for informal and opportunistic meetings. Tech. Rep. GIT-GVU-99-06. Graphics, Visualization and Usability Center, Georgia Institute of Technology, Atlanta, GA.
- BUSH, V. 1945. As we may think. *Atlantic Mon.* 176, 1 (July), 101–108. Also at <http://www.theatlantic.com/unbound/flashbks/computer/bushf.htm>.
- CARD, S., MACKINLAY, J., AND SHNEIDERMAN, B. 1999. *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann Publishers Inc., San Francisco, CA.
- CARROLL, J. M. AND ROSSON, M. B. 1991. Deliberated evolution: Stalking the view matcher in design space articles. *Hum. Comput. Interact.* 6, 3-4, 281–318.
- CHEVERST, K., MITCHELL, K., AND DAVIES, N. 1998. Design of an object model for a context sensitive tourist guide. In *Proceedings of the Conference on Interactive Applications of Mobile Computing* (IMB '98, Rostock, Germany, Nov.),
- CHIU, P. AND WILCOX, L. 1998. A dynamic grouping technique for ink and audio notes. In *Proceedings of the 11th Annual ACM Symposium on User Interface Software and Technology*

- (UIST '98, San Francisco, CA, Nov. 1–4), E. Mynatt and R. J. K. Jacob, Eds. ACM Press, New York, NY, 195–202.
- DAVIS, R. C., LANDAY, J. A., CHEN, V., HUANG, J., LEE, R. B., LI, F. C., LIN, J., MORREY, C. B., III, SCHLEIMER, B., PRICE, M. N., AND SCHLIT, B. N. 1999. NotePals: Lightweight note sharing by the group, for the group. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI '99, Pittsburgh, PA, May), ACM Press, New York, NY, 338–345.
- EDWARDS, W. K. AND MYNATT, E. D. 1997. Timewarp: techniques for autonomous collaboration. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI '97, Atlanta, GA, Mar. 22–27), S. Pemberton, Ed. ACM Press, New York, NY, 218–225.
- ELROD, S., BRUCE, R., GOLD, R., GOLDBERG, D., HALASZ, F., JANSSEN, W., LEE, D., MCCALL, K., PEDERSEN, E., PIER, K., TANG, J., AND WELCH, B. 1992. Liveboard: a large interactive display supporting group meetings, presentations, and remote collaboration. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI '92, Monterey, CA, May 3–7), P. Bauersfeld, J. Bennett, and G. Lynch, Eds. ACM Press, New York, NY, 599–607.
- FEINER, S., MACINTYRE, B., AND SELIGMANN, D. 1993. Knowledge-based augmented reality. *Commun. ACM* 36, 7 (July), 53–62.
- FERTIG, S., FREEMAN, E., AND GELEENTER, D. 1996. Lifestreams: an alternative to the desktop metaphor. In *Proceedings of the CHI '96 Conference Companion on Human Factors in Computing Systems: Common Ground* (CHI '96, Vancouver, British Columbia, Canada, Apr. 13–18), M. J. Tauber, Ed. ACM Press, New York, NY, 410–411.
- FITZMAURICE, G. W., ISHII, H., AND BUXTON, W. A. S. 1995. Bricks: Laying the foundations for graspable user interfaces. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI '95, Denver, CO, May 7–11), I. R. Katz, R. Mack, L. Marks, M. B. Rosson, and J. Nielsen, Eds. ACM Press/Addison-Wesley Publ. Co., New York, NY, 442–449.
- FITZMAURICE, G. W., ZHAI, S., AND CHIGNELL, M. H. 1993. Virtual reality for palmtop computers. *ACM Trans. Inf. Syst.* 11, 3 (July 1993), 197–218.
- FRANKISH, C., JONES, D., AND HAPESHI, K. 1992. Decline in accuracy of automatic speech recognition as a function of time on task: Fatigue or voice drift?. *Int. J. Man-Mach. Stud.* 36, 6 (June 1992), 797–816.
- FRANKISH, C., HULL, R., AND MORGAN, P. 1995. Recognition accuracy and user acceptance of pen interfaces. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI '95, Denver, CO, May 7–11), I. R. Katz, R. Mack, L. Marks, M. B. Rosson, and J. Nielsen, Eds. ACM Press/Addison-Wesley Publ. Co., New York, NY, 503–510.
- GOLDBERG, D. AND RICHARDSON, C. 1993. Touch-typing with a stylus. In *Proceedings of the ACM Conference on Human Factors in Computing* (INTERCHI '93, Amsterdam, The Netherlands, Apr. 24–29), S. Ashlund, A. Henderson, E. Hollnagel, K. Mullet, and T. White, Eds. ACM Press, New York, NY, 80–87.
- GRUDIN, J. AND PALEN, L. 1997. Emerging groupware successes in major corporations: Studies of adoption and adaptation. In *Worldwide Computing and Its Applications*, T. Masuda, Y. Masunaga, and M. Tsukamoto, Eds. Springer Lecture Notes in Computer Science, vol. 1274. Springer-Verlag, New York, 142–153.
- HARRISON, B. L., FISHKIN, K. P., GUJAR, A., MOCHON, C., AND WANT, R. 1998. Squeeze me, hold me, tilt me! An exploration of manipulative user interfaces. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI '98, Los Angeles, CA, Apr. 18–23), M. E. Atwood, C.-M. Karat, A. Lund, J. Coutaz, and J. Karat, Eds. ACM Press/Addison-Wesley Publ. Co., New York, NY, 17–24.
- HENDERSON, D. A. AND CARD, S. 1986. Rooms: The use of multiple virtual workspaces to reduce space contention in a window-based graphical user interface. *ACM Trans. Graph.* 5, 3 (July 1986), 211–243.
- HINDUS, D. AND SCHMANDT, C. 1992. Ubiquitous audio: capturing spontaneous collaboration. In *Proceedings of the ACM Conference on Computer-Supported Cooperative Work* (CSCW '92, Toronto, Canada, Oct. 31–Nov. 4), M. Mantel and R. Baecker, Eds. ACM Press, New York, NY, 210–217.

- ISHII, H. AND ULLMER, B. 1997. Tangible bits: Towards seamless interfaces between people, bits and atoms. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI '97, Atlanta, GA, Mar. 22–27), S. Pemberton, Ed. ACM Press, New York, NY, 234–241.
- JEBARA, T., EYSTER, C., WEAVER, J., STARNER, T., AND PENTLAND, A. 1997. Stochasticks: Augmenting the billiards experience with probabilistic vision and wearable computers. In *Proceedings of the IEEE International Symposium on Wearable Computing* (Oct.), IEEE Computer Society, New York, NY.
- LAMMING, M. AND FLYNN, M. 1994. “Forget-me-not” intimate computing in support of human memory. Tech. Rep. EPC-94-103. Rank Xerox, EuroPARC, Cambridge, UK.
- MACINTYRE, B. AND FEINER, S. 1996. Future multimedia user interfaces. *Multimedia Syst.* 4, 5, 250–268.
- MACINTYRE, B. AND MYNATT, E. D. 1998. Augmenting intelligent environments: Augmented reality as an interface to intelligent environments. In *Intelligent Environments Symposium* (Stanford, CA, Mar. 23–25), AAAI 1998 Spring Symposium Series AAAI Press, Menlo Park, CA.
- MANKOFF, J., HUDSON, S., AND ABOWD, G. 2000. Providing integrated toolkit-level support for ambiguity in recognition-based interfaces. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI '00, The Hague, The Netherlands, Apr.), ACM Press, New York, NY, 368–375.
- MARX, M. AND SCHMANDT, C. 1994. Putting people first: Specifying proper names in speech interfaces. In *Proceedings of the 7th Annual ACM Symposium on User Interface Software and Technology* (UIST '94, Marina del Rey, CA, Nov. 2–4), P. Szekely, Ed. ACM Press, New York, NY, 29–37.
- MINNEMAN, S., HARRISON, S., JANSSEN, B., KURTENBACH, G., MORAN, T., SMITH, I., AND VAN MELLE, B. 1995. A confederation of tools for capturing and accessing collaborative activity. In *Proceedings of the 3rd International Conference on Multimedia* (Multimedia '95, San Francisco, CA, Nov. 5–9), P. Zellweger, Ed. ACM Press, New York, NY, 523–534.
- MORAN, T. P., CHIU, P., HARRISON, S., KURTENBACH, G., MINNEMAN, S., AND VAN MELLE, W. 1996. Evolutionary engagement in an ongoing collaborative work process: A case study. In *Proceedings of the 1996 ACM Conference on Computer-Supported Cooperative Work* (CSCW '96, Boston, MA, Nov. 16–20), G. Olson, J. Olson, and M. S. Ackerman, Eds. ACM Press, New York, NY, 150–159.
- MORAN, T. P., CHIU, P., AND VAN MELLE, W. 1997a. Pen-based interaction techniques for organizing material on an electronic whiteboard. In *Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology* (UIST '97, Banff, Alberta, Canada, Oct. 14–17), G. Robertson and C. Schmandt, Eds. ACM Press, New York, NY, 45–54.
- MORAN, T. P., PALEN, L., HARRISON, S., CHIU, P., KIMBER, D., MINNEMAN, S., VAN MELLE, W., AND ZELLWEGER, P. 1997b. “I’ll get that off the audio”: A case study of salvaging multimedia meeting records. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI '97, Atlanta, GA, Mar. 22–27), S. Pemberton, Ed. ACM Press, New York, NY, 202–209.
- MUKHOPADHYA, S. AND SMITH, B. 1999. Passive capture and structuring of lectures. In *Proceedings of the ACM Conference on Multimedia* (Multimedia '99, Orlando, FL), ACM Press, New York, NY, 477–487.
- MYNATT, E. D. 1999. The writing on the wall. In *Proceedings of the 7th IFIP Conference on Human-Computer Interaction* (INTERACT '99), IFIP, Laxenburg, Austria.
- MYNATT, E. D., BACK, M., WANT, R., BAER, M., AND ELLIS, J. B. 1998. Designing audio aura. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI '98, Los Angeles, CA, Apr. 18–23), M. E. Atwood, C.-M. Karat, A. Lund, J. Coutaz, and J. Karat, Eds. ACM Press/Addison-Wesley Publ. Co., New York, NY, 566–573.
- MYNATT, E. D., IGARASHI, T., EDWARD, W. K., AND LAMARCA, A. 1999. Flatland: New dimensions in office whiteboards. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI '99, Pittsburgh, PA, May), ACM Press, New York, NY.
- NORMAN, D. A. 1998. *The Invisible Computer*. MIT Press, Cambridge, MA.

- OPPERMANN, R. AND SPECHT, M. 1998. Adaptive support for a mobile museum guide. In *Proceedings of the Conference on Interactive Applications of Mobile Computing* (IMB '98, Rostock, Germany, Nov.), Available at <http://www.egd.igd.fhg.de/imc98/proceedings.html>.
- PICARD, R. W. 1997. *Affective Computing*. MIT Press, Cambridge, MA.
- REKIMOTO, J. 1999. Time-machine computing: A time-centric approach for the information environment. In *Proceedings of the ACM Symposium on User Interface Software Technology* (UIST '99, Asheville, NC, Nov.), ACM, New York, NY, 45–54.
- REKIMOTO, J. AND NAGAO, K. 1995. The world through the computer: computer augmented interaction with real world environments. In *Proceedings of the 8th ACM Symposium on User Interface and Software Technology* (UIST '95, Pittsburgh, PA, Nov 14–17), G. Robertson, Ed. ACM Press, New York, NY, 29–36.
- RHODES, B. J. 1997. The wearable remembrance agent: A system for augmented memory. *Pers. Tech. J.* 1, 4, 218–224.
- RHODES, B. J. AND STARNER, T. E. 1996. Remembrance agent: A continuously running automated information retrieval system. In *Proceedings of Practical Applications of Intelligent Agents and Multi-Agent Technology* (PAAM),
- RICHTER, H., SCHUCHHARD, P., AND ABOWD, G. D. 1999. Automated capture and retrieval of architectural rationale. In *Online Proceedings of the 1st Working IFIP Conference on Software Architecture* (San Antonio, TX), IFIP, Laxenburg, Austria. Full paper appears as Georgia Tech GVU Center Technical Report GIT-GVU-98-37.
- SALBER, D., DEY, A. K., AND ABOWD, G. D. 1999. The context toolkit: Aiding the development of context-enabled applications. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI '99, Pittsburgh, PA, May), ACM Press, New York, NY, 434–441.
- SCHMANDT, C. AND ARONS, B. 1985. PhoneSlave: A graphical telecommunications interface. *Proc. Soc. Inf. Display* 26, 1, 79–82.
- SCHOMAKER, L. R. B. 1994. User-interface aspects in recognizing connected-cursive handwriting. In *Proceedings of the IEE Colloquium on Handwriting and Pen-Based Input*, The Institution of Electrical Engineers, London, UK.
- SHIPMAN, F. M., MARSHALL, C. C., AND MORAN, T. P. 1995. Finding and using implicit structure in human-organized spatial layouts of information. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI '95, Denver, CO, May 7–11), I. R. Katz, R. Mack, L. Marks, M. B. Rosson, and J. Nielsen, Eds. ACM Press/Addison-Wesley Publ. Co., New York, NY, 346–353.
- STARNER, T., MANN, S., RHODES, B., LEVINE, J., HEALEY, J., KIRSCH, D., PICARD, R., AND PENTLAND, A. 1997. Augmented reality through wearable computing. *Presence: Teleoper. Virtual Environ.* 6, 4.
- STIFELMAN, L. J. 1996. Augmenting real-world objects: a paper-based audio notebook. In *Proceedings of the CHI '96 Conference Companion on Human Factors in Computing Systems: Common Ground* (CHI '96, Vancouver, British Columbia, Canada, Apr. 13–18), M. J. Tauber, Ed. ACM Press, New York, NY, 199–200.
- STIFELMAN, L. J. 1997. The audio notebook: Paper and pen interaction with structured speech. Ph.D. Dissertation. MIT Media Lab, Cambridge, MA.
- STREITZ, N. A., GEISLER, J., HOLMER, T., KONOMI, S. I., MÜLLER-TOMFELDE, C., REISCHL, W., REXROTH, P., SEITZ, P., AND STEINMETZ, R. 1999. i-Land: An interactive landscape for creativity and innovation. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI '99, Pittsburgh, PA, May), ACM Press, New York, NY, 120–127.
- TRUONG, K. N., ABOWD, G. D., AND BROTHERTON, J. A. 1999. Personalizing the capture of public experiences. In *Proceedings of the ACM Symposium on User Interface Software Technology* (UIST '99, Asheville, NC, Nov.), ACM, New York, NY, 121–130.
- TURK, M., Ed. 1997. *Proceedings of the Perceptual User Interfaces Workshop*. (PUI '97, Banff, Alberta, Canada, Oct. 20–21). <http://research.microsoft.com/PUIWorkshop97/>.
- TURK, M., Ed. 1998. *Proceedings of the Perceptual User Interfaces Workshop*. (PIU '98, San Francisco, CA, Nov. 5–6). <http://research.microsoft.com/PUIWorkshop/PUI98.htm>.
- VAN BUSKIRK, R. AND LALOMIA, M. 1995. The just noticeable difference of speech recognition accuracy. In *Conference Companion for the ACM Conference on Human Factors in Comput-*

- ing Systems* (CHI '95, Denver, CO, May 7–11), J. Miller, I. Katz, R. Mack, and L. Marks, Eds. ACM Press, New York, NY, 95.
- WANT, R., HOPPER, A., FALCÃO, V., AND GIBBONS, J. 1992. The active badge location system. *ACM Trans. Inf. Syst.* 10, 1 (Jan. 1992), 91–102.
- WANT, R., SCHILIT, B., ADAMS, N., GOLD, R., PETERSEN, K., ELLIS, J., GOLDBERG, D., AND WEISER, M. 1995. The PARCTab ubiquitous computing experiment. Tech. Rep. CSL-95-1. Xerox PARC, Palo Alto, CA.
- WEHER, K. AND POON, A. 1994. Marquee: A tool for real-time video logging. In *Proceedings of the ACM Conference on Human Factors in Computing Systems: "Celebrating Interdependence"* (CHI '94, Boston, MA, Apr. 24–28), ACM Press, New York, NY, 58–64.
- WEISER, M. 1991. The computer for the 21st century. *Sci. Am.* 265, 3 (Sept.), 94–104.
- WEISER, M. 1993. Some computer science issues in ubiquitous computing. *Commun. ACM* 36, 7 (July), 75–84.
- WELLNER, P. 1993. Interacting with paper on the DigitalDesk. *Commun. ACM* 36, 7 (July), 87–96.
- WHITE, S., GUPTA, A., GRUDIN, J., KIMBERLY, G., AND SANOCKI, E. 1998. A software system for education at a distance: Case study results. Microsoft Research Tech. Rep. MSR-TR-98-61. Microsoft Corp., Redmond, WA. Revised in April 1999.
- WHITTAKER, S., HYLAND, P., AND WILEY, M. 1994. FILOCHAT: handwritten notes provide access to recorded conversations. In *Proceedings of the ACM Conference on Human Factors in Computing Systems: "Celebrating Interdependence"* (CHI '94, Boston, MA, Apr. 24–28), ACM Press, New York, NY, 271–277.
- WILCOX, L. D., SCHILIT, B. N., AND SAWHNEY, N. 1997. Dynamite: a dynamically organized ink and audio notebook. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI '97, Atlanta, GA, Mar. 22–27), S. Pemberton, Ed. ACM Press, New York, NY, 186–193.
- WOLF, C. G., RHYNE, J. R., AND BRIGGS, L. K. 1992. Communication and information retrieval with a pen-based meeting support tool. In *Proceedings of the ACM Conference on Computer-Supported Cooperative Work* (CSCW '92, Toronto, Canada, Oct. 31–Nov. 4), M. Mantel and R. Baecker, Eds. ACM Press, New York, NY, 322–329.
- ZAJICEK, M. AND HEWITT, J. 1990. An investigation into the use of error recovery dialogues in a user interface management system for speech recognition. In *Proceedings of the IFIP 1990 Conference on Human-Computer Interaction* (INTERACT '90), IFIP, Laxenburg, Austria, 755–760.

Received: February 1999; revised: September 1999 and February 2000; accepted: February 2000