Carnegie Mellon University
Research Showcase @ CMU

Human-Computer Interaction Institute | School of Computer Science

2007

Momento: Support for Situated Ubicomp Experimentation

Scott Carter
FX Palo Alto Laboratory

Jennifer Mankoff
Carnegie Mellon University

Jeffrey Heer
University of California - Berkeley

Follow this and additional works at: http://repository.cmu.edu/hcii

This Conference Proceeding is brought to you for free and open access by the School of Computer Science at Research Showcase @ CMU. It has been accepted for inclusion in Human-Computer Interaction Institute by an authorized administrator of Research Showcase @ CMU. For more information, please contact research-showcase@andrew.cmu.edu.

Momento: Support for Situated Ubicomp Experimentation

Scott Carter
FX Palo Alto Laboratory
3400 Hillview Avenue, Bldg 4
Palo Alto, CA 94304
carter@fxpal.com

Jennifer Mankoff
HCI Institute
Carnegie Mellon University
Pittsburgh, PA 15213
jmankoff@acm.org

Jeffrey Heer
BID and CSD
University of California, Berkeley
Berkeley, CA 94720
jheer@cs.berkeley.edu

ABSTRACT

We present the iterative design of Momento, a tool that provides integrated support for situated evaluation of ubiquitous computing applications. We derived requirements for Momento from a user-centered design process that included interviews, observations and field studies of early versions of the tool. Motivated by our findings, Momento supports remote testing of ubicomp applications, helps with participant adoption and retention by minimizing the need for new hardware, and supports mid-to-long term studies to address infrequently occurring data. Also, Momento can gather log data, experience sampling, diary, and other qualitative data.

Author Keywords

Mobile Devices, Wizard of Oz, Rapid Prototyping, Experience Sampling, Diary Studies

ACM Classification Keywords

H5.2 [Information interfaces and presentation]: User Interfaces. - Graphical user interfaces. **General Terms:** Design, Human Factors, Experimentation

INTRODUCTION

Ubicomp applications are designed to support tasks that scale across multiple people, activities, and places. This often leads to an interdependence between applications and their surroundings that creates a need for real-world, situated approaches to iterative design. However, it is challenging to gather realistic data during the early stages of iterative design of ubicomp applications [5].

In interviews with nine developers of mobile technology [5], and observations of four diary studies (a type of study in which participants keep a journal about events of interest to the experimenter) [4], we identified the following difficulties with situated evaluation: remote testing; adoption and retention; and the need to study events that occur infrequently. Also, our interviews and observations highlighted the value of both quantitative data, such as logs and timestamps, and qualitative data, such as that gathered from experience sampling studies (ESM) and diary studies, as well as integrated tools for annotating and reviewing data.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
CHI 2007, 28 April - 3 May, 2007, San Jose, California, USA.
Copyright 2007 ACM 1-59593-178-3/07/0004...$5.00.

In this paper, we present the user-centered development of Momento, a tool that provides integrated support for situated evaluation of ubicomp applications and that addresses the key issues raised in our interviews and observations. In particular, Momento can piggyback on participants' existing networked mobile devices to minimize the need for new hardware and reduce issues of adoption and retention. Momento supports remote testing and can support mid-to-long term studies to address infrequently occurring data. Experimenters can use Momento to gather quantitative log data, ESM, diary, and other qualitative data. Momento is designed to be easy to use, and can be configured and run by installing a mobile client and associated text-based configuration file on participants' devices and using a GUI desktop platform to configure experimental details. Momento supports monitoring incoming information from and sending information to participants, and reviewing data or exporting it for analysis.

Momento's initial requirements came from our interviews and observations. As we developed Momento, we deployed it in several pilots and four studies involving ourselves and external researchers, ranging from two days to two months in length and including from four to fourteen users. After each study, we analyzed experimenter and participant experiences when using Momento and used this information to further refine our design. In particular, studies led to changes to supported mobile platforms and networking protocols, augmented support for mobile experimenters and automated sampling, and privacy-sensitive support for *post hoc* annotation and interviews.

Momento leverages the text messaging (SMS) and media messaging (MMS) capabilities of mobile devices such as mobile phones to share information between end users and experimenters, who use a desktop platform designed for managing experiments. Using Momento, experimenters can respond to participant requests, ask participants to manually capture or record data, or automatically gather data from mobile devices. Today's media-enabled mobile devices are relatively ubiquitous, and have the ability to capture and transfer a variety of media and data, ranging from Bluetooth and GPS data to audio and images. It is estimated that 785 million camera phones will be shipped worldwide in 2010, comprising 87% of all mobile phones [18].

Since our contributions center on the iterative, user-centered design of usable support for situated experimentation, so do our innovations. By situated experimentation, we mean that Momento's goal is to allow experimentation that takes into account the context of use. Momento provides experimen-

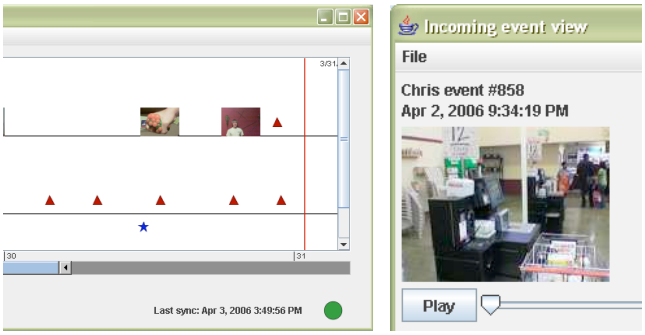

Figure 1: (left) Part of an overview display for use by remote experimenters while a study is running. This display is especially valuable during lengthy studies with sparse data. **(right)** Part of a detailed view of a message.

ters with an overview display to help them manage lengthy studies with sparse data (shown in Fig. 1, left). Paired with that, Momento also enables context-sensitive, real-time notifications to be sent to experimenters based on information sensed about users. A configurable mobile client suitable for many experiments is provided, and a subset of experiments can be carried out without any client-side installation. Finally, by integrating support for collecting quantitative and qualitative data, we provide a consistent end user experience.

Overview

The next section presents our interviews and observations and the requirements arising from them. We then describe the current version of Momento, after eighteen months of iterative design involving pilots and four studies. One study is used to illustrate this description. The other three studies, which demonstrate different experimental paradigms, are presented in more depth in our validation section. We report on the end user and experimenter experience and conclude by highlighting areas needing development.

FIELDWORK

Our investigation into situated evaluation of ubicomp technology began with two pieces of work: First, we observed two different sets of experimenters conducting diary studies and conducted two diary studies ourselves [4]. Second, we interviewed nine developers engaged in the iterative design of mobile ubicomp applications [5]. Here we unify those pieces of work and focus on their implications for the design of a tool intended to support situated experimentation.

Observations of Diary Studies

The primary goal of our diary study work was to understand the impact of the ability to easily gather a range of media (such as location, photos, and audio) on the types of information that experimenters could elicit from subjects [4]. However, as we also reported, it quickly became clear that the uses of these media led to a need for better tools.

Even though diary studies are well suited to gathering situated data, when participants were asked to record photos or audio, they could not always explain or remember what they chose to record ("that could have been anywhere," one participant said of one of his audio recordings, "maybe the weird wind by the train? I don't know."). Additional context such as location or time could help with this. The tools typically given to participants sometimes lacked basic features such as time-stamping and did not support more complex features such as lightweight, situated annotation of captured media.

The experimenters we observed often chose only to download data from capture devices at the start of interviews to minimize participant burden. This practice led to some problems. Experimenters did not have time to review captured data before elicitation interviews, which curtailed their ability to prepare for the interview and increased the chance that important themes would be missed. Some of the media captured by users did not align with the goals of the study, and in one case experimenters did not address this until the experiment ended. Being able to view user captures in real time could help with feedback and could also improve experimenters' ability to prepare for elicitation interviews.

While our observations of diary studies highlighted some of the problems encountered with a particular experimental technique, the interviews described next showed us some more general problems facing experimenters engaged in an iterative design process.

Interviews with Mobile Developers

We conducted interviews with nine developers of mobile technologies. All nine interviewees had conducted at least one field experiment. Interviewees had explored a range of data gathering techniques including diary studies and interaction logs. Interviewees preferred to conduct usability tests remotely, both to avoid inhibiting normal behavior and because of the cost of being "everywhere at the same time" [27]. In all but one case, only log data was gathered during remote evaluations, and it was hard to understand the context of use for that data. One interviewee used event-contingent ESM and reported that this was highly effective. In this method, queries to participants are triggered by context, to focus feedback on moments when something important occurs. The interviewee sent a query to participants every time they interacted with his application. This enabled him to gather more than only log data and made the evaluation worthwhile. However, in general, interviewees reported several problems that made it hard to conduct successful situated evaluations, described next.

Adoption and retention of new technologies, even for the length of a single study, were major problems. For users who already own mobile devices, adding or switching devices was a big barrier to use. Although the developers we interviewed wished to leverage existing devices, they were not able to do so in their field studies. As a result, they had to work hard "staying on top of users" to encourage them to continue using deployed devices and to report problems in a timely fashion, rather than simply abandoning their devices.

A third problem was gathering enough data. Some tasks may naturally occur only occasionally (such as commuting to and from work), or may be difficult to sense (such as an emotional response). For example, two of our interviewees conducted field studies in which the events they were interested in happened only twice per day. This led them to depend more heavily on qualitative data such as interviews and experience sampling, made it harder to collect the quantitative data they would have liked, and pushed them towards longer studies.

Summary of Requirements: Participants

Our fieldwork helped us to identify usability requirements for participants and for experimenters. For participants, minimizing the burden of experimentation can help with participant compliance, adoption, and retention. We identified three main issues that can be addressed:

Unified client system A unified client system for gathering multiple types of data (such as self reports, logged events, and ESM) is one way of increasing consistency and mitigating adoption and retention issues raised in our interviews.

Leverage existing devices Our field work confirmed that carrying a new device, even loaded with useful new features, is burdensome for end users. It is important to leverage existing devices as much as possible. Communications with end users can piggyback on commonly available devices such as camera phones and devices that support text messaging.

Multiple, lightweight communication options Related to this, it is important to provide multiple kinds of support for communication between participant/device and experimenter. Our diary study observations showed the need for ongoing availability of communication and asynchronous communication (*e.g.*, annotation); and both our diary studies and interviews emphasized the need for communication to be lightweight. Given a focus on participants' existing devices, options include: live phone discussions on a participant's phone (hard to manage asynchronously but allows rapid communication); text messaging (ideal for lightweight experimentation with participants who are highly comfortable with text messaging); or a custom client (requires software installation, is easily customizable by the experimenter, can include many more types of media and can be optimized for participant usability).

Summary of Requirements: Experimenters

For experimenters, it is important to support different levels of computer experience, to support multiple types of data and context, and to support monitoring and notification. Some specific requirements arising from our studies include:

Support qualitative data, quantitative data, and context Experimenters reported valuing both qualitative and quantitative data, including diary data, ESM data, log data, and sensed context. Integrating support for multiple types of data can help to increase consistency and usability.

Do not require fully implemented applications Participants in our observations wanted to test ideas they could not easily implement. To support experimentation at the early stages of design, as well as to support experimenters with limited coding experience, it is important not to require complete applications. One way to facilitate this is to support a Wizard of Oz protocol in which experimenters can do some of the work normally done by an application.

Support the full experimental lifecycle The experimental lifecycle described by our interviewees included experimental set-up, modifications to an experiment, running an experiment, and analysis and summarization of data both during and after the experiment was run. A usable system should support this entire cycle.

Support monitoring and notification As reported and observed in our field work, experimental data may arrive in occasional bursts, reflecting the variable nature of day to day life. When a study takes place over days or weeks, constant attention may be an inefficient, difficult way to watch for rare or uneven data. Overview displays and notifications can address this problem. For example, an experimenter might be notified when a user enters a certain space, or might monitor a display for big changes in amount of activity. These pieces of information could help an experimenter decide when to take direct action (*e.g.*, contact the user, go somewhere to make observations, *etc.*).

Support lengthy, remote studies Our fieldwork identified a preference for remote data gathering. Furthermore, data from studies that take place over days, weeks, or more may overcome issues such as the novelty effect and learning. Gathering situated data over time is more feasible when experimenters can be remote. Additionally, remote experimentation helps to mitigate the effects of an observer being present.

Based on these requirements, we created a tool, Momento, supporting situated experimentation. Our process involved iterative studies with the tool, run by ourselves and others. After each study, we reflected on the participant and experimenter experience. The current version of Momento is described next, while studies run during and after the implementation of Momento are described at the end of this paper.

SYSTEM DESCRIPTION

Momento consists of a set of configurable tools that can be used without writing source code – our goal is that anyone with basic computer and mobile device experience will be able to use the tools. As shown in Fig. 2, the most important components of Momento are the clients (C) used primarily by participants to send messages and make requests (clients include fixed applications and mobile devices running either standard mobile multimedia applications or a specialized mobile application we built); the desktop platform used by experimenters to configure and monitor experiments (D); and the server (S), which supports multiple experiments, handles communication between clients and the desktop for each experiment, and provides remote access facilities.

All communication in Momento is sent as text or multimedia messages. Messages are sent to and from the server and mobile devices or fixed applications using HTTP or SMS/MMS (text or multimedia messaging). Momento communicates with networked applications using the Context Toolkit (CTK) [9]: messages are sent to applications using the CTK event system and are sent back to Momento using the CTK services system.

Clients

Participants, not experimenters, are the primary users of clients. The most basic client facilitates data gathering, such as requesting a photo or implicitly logging sensed context. It is also possible to simulate an interactive application through the mobile client. Mobile experimenters can also use clients to report context information from the field or to interact with participants while mobile.

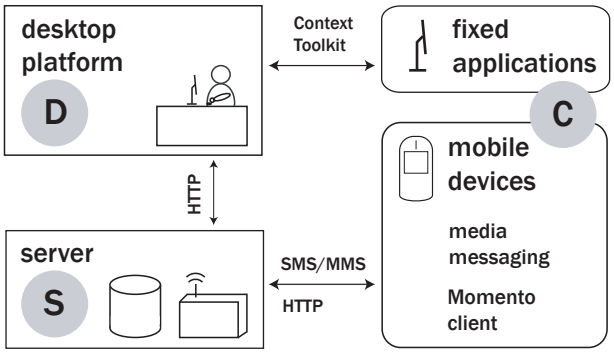

Figure 2: System architecture. The desktop platform (D) and server (S) communicate with clients (C) via SMS/MMS, HTTP, or the Context Toolkit. Study data is stored for later analysis or retrieval via the web.

Listing 1: The structure of the client configuration file

```
# Description of study (Any string)
Scribe4Me study
# Participant ID (Any string)
John
# Server location (IP)
236.345.2.1
# Continuously buffered data types
# (name and buffer length in secs)
# audio|bluetooth|photo|gps [30|60|80]
audio 30
# Button definitions:
# name [color] label [media]* [annotation?]
# [name],[HSV],[string],
# [audio|bluetooth|photo|gps|text]*,[annotate]
what, ``what happened?'', audio, photo
```

Data gathering The main purpose of Momento is to support situated experimental data gathering. An important part of this is qualitative reports from users. In addition to SMS/MMS, Momento includes an easily configurable mobile client that can be installed on most mobile devices. This client can display an information request to a user, who can then respond by taking a photo (if supported by the device), recording audio, entering text, or sketching using a stylus.

Context Both the mobile client and fixed applications can report on context data. The mobile client supports logging of location, nearby people, and audio.

Simulating applications In addition to information requests, arbitrary messages may be sent to clients independently or in response to user input. This can be used to simulate an interactive application. By default, the Momento mobile client displays incoming messages to the participant (more sophisticated behavior must be hand-coded, as must any behavior by fixed applications). Also, the client may be configured to display one or more buttons that a user can select. Depending on its configuration, a button may send locally cached data to the desktop platform or may request media from the user and send it (*e.g.*, ask the user to enter text or take a photo).

Mobile experimenters Although we support experimental protocols that do not require an experimenter to be mobile, there are still occasions when mobility is desirable. For example, a mobile experimenter might wish to observe participants or to record data that is hard to sense. Mobile experimenters in Momento can use the mobile client to report data or to view live data recorded by participants. In particular, messages can be forwarded to a mobile experimenter who can optionally respond to participants. Finally, Momento can notify an experimenter when a participant is in a certain context (*e.g.*, in a certain location with a group of people). The experimenter may wish to act on this information, for example by making additional observations in person.

Versions of Momento's mobile client are implemented in Java and in C#. The mobile client can receive messages via SMS/MMS from the server and can send messages back using either SMS/MMS or HTTP over a network connection, if one is available. In this way, clients on mobile devices can leverage the fastest network connection available to them (*e.g.*, 802.11, GPRS, TCP/IP over Bluetooth, *etc.*) to send messages to the server, which will pass them on to the experimenter's desktop platform. The mobile client buffers state so that it is robust to network and application interruptions (*e.g.*, loss of connectivity, phone calls, *etc.*).

The mobile client is configured using a simple text file. Listing 1 shows the basic structure of the configuration file. The experimenter can instruct the client to upload specific media to the Momento desktop platform either when a button is pressed, as soon as information is detected (for discrete events) or at certain intervals.

Listing 1 corresponds exactly to the Scribe4Me application (Fig. 3) [24]. Audio is stored in a 30-second buffer, and the interface contains a single button (labeled "what happened?"). When the participant presses the button, the application asks that a photo be taken and then transmits the photo and current audio buffer to the desktop platform.

Desktop Platform

The desktop platform is the main interface used by experimenters. It supports many of the activities of the experimental lifecycle. Experimenters can use it to manage the specification of the devices, participants, groups, locations, and rules associated with the experiment. While the experiment is running, experimenters can see an overview of communications with the participants that supports peripheral monitoring, manage communications by sending messages and viewing incoming messages, and receive notifications about actions that are needed. The desktop platform sends text and media messages to mobile devices with the server's help, and communicates with other client applications via the Context Toolkit. All incoming and outgoing messages are stored locally by the desktop platform. Experimenters may upload this data to the server at will. In this section we describe each supported activity.

Specification of the experiment Specification, which typically takes place during or just after the planning phases of an experiment, includes the devices, participants, groups (of participants), locations, and rules.

> **Participants and groups** Participants are uniquely defined by their client ID (typically a mobile phone number). Other information about participants such as client capabilities, demographics, and a calendar schedule may also be recorded. Experimenters can create groups

if [cond and/or cond ...]	*then send* [content]	*to* [recipient(s)]
o/g near/!near location o/g arrives/leaves location o/g stays in location (min time) msg arrives msg has attachment msg matches regular expression time of day (repeat time)	file (random|specific) [o's name] [o's location] [o's proximate people] [incoming message] arbitrary text web page	any person any group

Table 1: Rules. o = originator of the message; g = group. In the conditional, g is matched if anyone in the group is the originator.

of participants for use when rules or messages should apply to more than one person.

Locations The IDs of Bluetooth beacons can be recorded in the desktop platform to support discrete semantic locations (*e.g.*, near a landmark or other phone). Furthermore, Momento can use standard serial protocols to communicate with GPS devices to support geographically defined locations (GPS coordinates).

Rules Experimenters can pre-configure outgoing messages using Momento's rule system. The rules system can be used to automate simple, frequent actions and to configure ESM and event-contingent ESM studies. Experimenters can configure Momento to evaluate rules at specific times or periodically (*e.g.*, "evaluate this rule if it is 3pm and participant Chris is in his office"). In addition, each rule in Momento is evaluated when a message arrives. Rules take the form: `if` [conditional and/or conditional] `then send` [content] `to` [recipient(s)] (see Table 1).

Overview and management of communications The desktop platform's overview mode, shown in Fig. 1 (left), displays recent messages sent to and from all participants. The timeline, built using prefuse [14], associates one horizontal line with each participant. Red triangles, or photos (if part of the message), represent incoming messages, while blue stars represent outgoing messages. A vertical red line indicates the current time. Blue stars shown past this line indicate future messages scheduled using the rule system. Clicking on a message icon opens an information panel showing message details and associated media (see Fig. 1, right).

Notifications Notifications are provided automatically when messages arrive to help experimenters monitor incoming messages. Notifications can allow an experimenter to multitask and can reduce the impact of experimenter fatigue. They are especially helpful when data are sparse. Notifications are visual and auditory.

Server

The Momento server has three functions: it acts as the gateway between the desktop platform and the cellular network, (allowing the desktop platform to run on any networked machine); it manages study data; and it provides password protected web access to captured media.

Gateway The server can communicate video, images, audio, or text-based data to and from email addresses, default messaging applications on mobile phones, or the Momento mobile client. The server automatically compresses media, and sends messages to phones via a GPRS-enabled modem.

Figure 3: The Scribe4Me system caches 30 seconds of audio and sends it to an experimenter for transcription when the user presses the "what happened?" button **(left)**. When the experimenter completes the transcription, it is displayed **(right)** [24].

Study management The server keeps a database of all data associated with each study. This allows experimenters to stop and start the desktop platform and move between different experiments.

Web host Web access is hosted by the server but can be controlled with the desktop platform. Data for each participant is provided separately, and a per-participant password (provided by the experimenter) is required for access. The web interface is implemented using PHP.

Example

As an example, consider a study we recently ran of the Scribe4Me system (shown in Fig. 3), which allows people who are deaf to request a transcription of recent audio [24]. The goals of our experiment were to learn when and whether such functionality would be valuable to people with a range of hearing impairments. For this reason, our experimenters wanted a photo showing additional context relevant to each audio request. Also, they used a diary protocol to ask participants to answer a fairly lengthy series of questions about each transcription request. This was done at the end of each day, and depended on the ability to show a history of the day's requests to participants. Although our original study ran with an earlier version of Momento that did not support all of these features, here we describe how it would be done with the current version.

Because of Momento's ability to capture live context, including audio, and transmit it to experimenters, Scribe4Me requires no implementation, only configuration. In general, we have found this to be true of many studies that involve an application that can be defined simply by transferring information and media between participants and a human experimenter. The participant device is configured to buffer 30 seconds of live audio at all times. Additionally, a button, labeled "what happened?" is added that transmits that buffer to experimenters when pressed. This button also causes a pop-up window to appear, asking participants to take a photo illustrating the reason for the request.

To run this experiment with Momento, only 4 actions are needed: First, the experimenter downloads and installs the mobile client on each participant's mobile device. Second, the experimenter modifies a total of five lines of the configuration file (as shown in Listing 1). Third, the experimenter configures the desktop platform, specifying the IP of the server and the phone numbers and ID of each participant and enables web access. Fourth, the experimenter can optionally specify details about what context should be gathered (by editing the mobile client's configuration file) or add rules to the desktop platform describing when communications should take place automatically.

Once configuration is complete and the experiment is running, event flow would look like the following. Note that all communications from the participant to the experimenter go through the mobile client, to the server, to the desktop platform and all communications from the experimenter to the participant do the reverse, unless otherwise specified.

1. Scribe4Me participant presses "what happened"; sends audio and photo to experimenter.
2. Experimenter sees data and enters transcription, which is sent to the participant.
3. At the end of the day, the participant reviews all of her requests using a web interface provided by the server, and fills out a form answering a series of questions that are then sent to the experimenter.
4. Optionally, an experimenter could also send a short-answer ESM request to the participant immediately after completing a transcription or at random times during the day. The participant's response would be shown immediately to the experimenter.

Summary

Momento supports situated ubicomp experimentation by managing, recording, and automating different types of data flow between participants, experimenters, and applications. However, our findings led us away from synchronous communication support. Our validation was conducted as part of an iterative design process. We deployed our system internally and with external experimenters as we developed it. Each time, we interviewed experimenters and end users, and analyzed the data that experimenters gathered. Our focus was on understanding and improving both the end user experience and the experimenter experience.

VALIDATION

Table 2 shows an overview of four experiments run using Momento. External researchers ran two of them: a diary study of young adults' approaches to informal learning (InformalLearning), and a mobile sketch-based learning application for children (PhotoSketch). We ran the other two in collaboration with other researchers: a public awareness display (AwarenessBoard), and the Scribe4Me system. These experiments demonstrate Momento's ability to support a variety of experimental methods, a variety of data collection techniques, and a range of study lengths. AwarenessBoard was run with an early version of Momento, while Scribe4Me was run about five months later with an intermediate version, and the other two were run when Momento was nearly in its current state. We iterated after each one.

Study name	Study type	Number of participants	Days per participant	Number of experimenters	Wizard of Oz	Feedback	Questions	Desktop review	Other applications	Mobile experimenters	Distributed experimenters	Desktop real-time	Mobile client	Rules
Informal learning	diary	12	7	4		■		■			■	■		■
PhotoSketch	workshop	24	1	7				■					■	
Scribe4Me	field study	6	7-15	2	■					■	■	■	■	■
AwarenessBoard	field study	14	51	3	■		■		■	■	■	■	■	■

Table 2: How experimenters used Momento in four different experiments, including whether or not the experiments involved Wizard of Oz, feedback or questions sent by experimenters to participants, use of the desktop to review captures with participants after the experiment, external applications, distributed or mobile experimenters, monitoring of the desktop during the experiment, the mobile client, or rules.

The experiments spanned a range of methods and experimenter conditions (see right columns in Table 2). AwarenessBoard and Scribe4Me were deployed as field studies lasting weeks to months, while PhotoSketch was deployed during 1-day workshops and the InformalLearning study was a week-long diary study. Messaging between experimenters and participants varied: Scribe4Me and AwarenessBoard depended on synchronous Wizard of Oz exchanges between experimenters and participants; the InformalLearning study involved the experimenters sending feedback to participants; and PhotoSketch required no messaging in the field. Furthermore, experimenters in all experiments monitored captured events in real-time, but those who ran the PhotoSketch and InformalLearning experiments also used Momento for *post hoc* review on the desktop platform with participants. Finally, in the AwarenessBoard experiment, Momento communicated directly with participants as well as other applications.

In this section, we describe the AwarenessBoard, Informal Learning and PhotoSketch experiments in detail. Our goal is to illustrate that we were successful in providing a usable experience to both end users and experimenters, as well as to illustrate a range of possible uses for Momento. For each experiment, we first provide a brief description of the experimenters' goals and describe the methods used. We then describe what we learned about both the participant and experimenter experience and how this facilitated the iterative design of Momento.

Early Study of AwarenessBoard

We used Momento to help implement an early prototype of a public display, the AwarenessBoard (shown in Fig. 4), intended to convey a history of the availability and location of participating faculty members in our department. The AwarenessBoard study was one of the earliest and most extensive studies run with Momento, and it contributed greatly to our understanding of the usability issues facing both participants and experimenters. It depended on almost all of Momento's features. The study lasted for two months and involved fourteen participants (including 12 faculty and two students). We were the primary experimenters, although we were collaborating with two social scientists who had designed an application to help test sociability and awareness in a distributed academic department.

Setup

The public display was designed to show faculty members' location availability, which was provided by Momento using the Context Toolkit. The mobile client was customized to show the same information. Sensing of location and availability was done using Bluetooth beacons in Momento. Availability was "sensed" using a simple heuristic that leveraged location information: If in a public place or alone in his or her office, a faculty member was available. If others were present, a faculty member was assumed to be in a meeting. Changes in sensed availability triggered SMS messages to faculty checking whether the estimation was correct. Faculty could optionally provide a calendar indicating times they did not wish to receive such interruptions. Experimenters monitored the desktop platform to ensure faculty were answering questions in a reasonable amount of time.

The availability heuristic and location sensing were implemented using Momento's rules system. Additionally, a rule was created that would notify one of our social scientist collaborators when someone interacted with a public display (*e.g.*, to view a faculty member's history). This allowed our collaborator to conduct in-person observations.

In total, we recruited twelve faculty participants to use the public display and answer questions using the mobile client. Faculty were each given mobile phones so they could respond to SMS messages confirming estimations of availability. The phones were in addition to their personal phones – there was too much variability in the technology they already carried to piggyback on existing devices. We also recruited two students who tested an SMS interface to the public display. These participants could send a message to Momento requesting the current availability of a faculty member. Responses to these requests were handled by a wizard monitoring the Momento desktop platform.

We ran pilots of the system for three weeks and ran the study over the course of two months. We iterated and improved both the public display and the client based on participant feedback over the course of the study, and made significant adjustments to the public display after the pilots and again during a holiday break six weeks into the main study.

Participant experience

Our participants were biased towards a population that rarely if ever used phone-based applications or SMS/MMS, and this led to some difficulties with the mobile client. Additionally, the unfamiliarity of the particular phone we gave to participants led to complaints that the phone was "big and bulky" or "stopped working" (the operating system on the phone we used notified users of irrelevant information and used modal dialogs to demand confirmations). As a result, users did not always keep the phone nearby. For these reasons, it was difficult to derive useful feedback for iterating the mobile client's interface. Primarily, this experience encouraged us to focus on improving our mobile client to piggyback on mobile devices that individual users already carry.

Experimenter experience

Regarding setup, implementation of the application being tested (shown in Fig. 4) was time consuming. However, linking it to the Momento infrastructure was straightforward. A bigger difficulty arose when setting up security for the Java-based mobile client during mobile device installation. To address this, we began working on a C# mobile client.

Figure 4: An awareness prototype deployed in a field setting. Location and availability of users were sensed via mobile clients running on users' mobile devices as well as experimenter input.

The day-to-day effort of running this field study varied. We found that experimenters controlling the mobile interface to the public display were able to monitor and respond to incoming questions with only minimal distractions from their other work. Also, experimenters wanted to observe participants using the public display without necessarily always having to be present. To support this, they suggested having Momento automatically generate messages that would be sent to experimenters' mobile devices whenever it was detected that a participant was using the display. Experimenters also wanted to be able to send messages to participants once they had been in a particular place for a certain amount of time. We added both of these features.

Because this study involved many different experimenters working from different locations, we had to increase the sophistication of the server-desktop platform networking. Initially, Momento handed-off messages from the server to the desktop platform through a networked folder, which required the desktop to be on the same network as the server.

Part way through the study, we added support for experimenters to walk through the hallways checking for participant availability and sending updates to Momento via mobile phones. This and other unanticipated models for participation by remote and mobile experimenters led us to expand Momento's sender-recipient model (who could send messages to whom) and to enhance the rule system to support time-based triggers and triggers from external applications connected via the Context Toolkit. With this change, experimenters were able to tie events triggered by external applications to messages sent to mobile experimenters. Finally, we modified networking between the server and mobile client to leverage HTTP. By supporting HTTP uploading, we could avoid slow media messaging communication when other data carriers (*e.g.*, 802.11b) were available.

Diary Study of Informal Learning

Four external experimenters used a recent version of Momento to conduct week-long diary studies of young adults' approaches to learning new technologies. The study was conducted at two sites over the course of two months: three

experimenters ran six participants at one site and two experimenters ran another six at the other site (one experimenter worked at both sites). All participants were 13-19 years old. Also, across both sites participants started and completed the study on different days. We interviewed experimenters about the data resulting from their study. Our goals in observing this study were to get feedback on the ability of Momento to support studies distributed across different sites and multiple experimenters as well as back-and-forth interaction between experimenter and participants during situated needs finding.

Setup

Participants used standard media messaging applications on their mobile phones to capture images. Participants unfamiliar with MMS were instructed to send text descriptions of events via SMS and, in some cases, given a digital camera. Later in the study, experimenters also occasionally sent feedback to participants to indicate that they needed to focus on capturing different events or simply to offer encouragement. At the end of the study, experimenters conducted in-person interviews with participants, using the Momento desktop platform to review captured data. The experimenters also reimbursed participants for any extra expenses incurred sending messages for the study.

Experimenters used the Momento server to share data across different sites. Typically, one researcher would run the desktop platform for a couple of days before handing off the study to another experimenter. To do this, an experimenter would use the Momento desktop platform to upload the study from her personal computer to the server, and a second experimenter would then use Momento to download the study from the server to his personal computer.

Participant experience

The study was in line with most participants' habits – they were comfortable taking photos with their phones. These participants often sent photos via media messaging to friends, and were able to use the same feature to send photos to Momento. This study helped to validate the advantages of piggybacking on familiar devices and applications.

Initially, participants were somewhat frustrated with the lack of immediate feedback from experimenters. Experimenters attempted to mitigate this concern by sending some feedback, positive or negative, for most captures. An unexpected side benefit of Momento seen in this study was that parents were pleased that their children could contribute to a study without being followed or watched by a stranger.

Experimenter experience

Momento facilitated many aspects of the experimental lifecycle. The experimenters were able to configure the study rapidly and run pilots using their own devices. They were then able to use the same settings (with the exception of the participants' mobile phone information) for the actual study. The experimenters then used the interface to monitor events during the study – in several cases they diagnosed problems with participants' mobile phones by noticing problems with captured data (*e.g.*, only text from a phone that supported MMS, or text added onto messages by the carrier network). As mentioned, experimenters also used the interface to provide feedback to participants, most commonly acknowledgment and encouragement.

Experimenters wanted to be able to use Momento during elicitation interviews. Specifically, they wanted to show individual participants their own timeline of images and to scroll through their events in the detail view. While this was already supported, it raised privacy concerns that we addressed: First, we augmented the timeline to allow experimenters to switch between views that included all participants versus only one participant. Second, we added the ability to scroll through incoming message details per-participant while keeping experimenter notes private. We also added a feature to print out physical copies of messages (including text and pictures) to support elicitation interviews at which a computer was not available.

Furthermore, experimenters felt that participants should be able to annotate media captures via the web, an issue that also arose in the Scribe4Me study. We added a web interface to the server with annotation support. Also, we observed that some experimenters had difficulty sifting through hundreds of captures represented on some participants' timelines. We added an interactive table to the desktop platform listing all incoming messages. Experimenters found the list a more useful way of navigating incoming messages than the timeline for periods of high response rates (more than 20 per participant-hour). Finally, experimenters imported Momento's data files into analysis programs.

PhotoSketch: Supporting Informal Classroom Learning

Another group of external researchers are using the current version of Momento to conduct one-day workshops exploring a mobile learning application (PhotoSketch) for children in fifth grade (ages 10-11) at an elementary school. The researchers have thus far conducted two workshops, involving 11 and 13 participants and three and four experimenters, respectively. Our primary goal for this experiment was to observe how easily Momento could be extended to different environments. We interviewed experimenters to understand the process of building a prototype application that integrated into a classroom scenario using Momento. We gathered feedback from the teachers as well.

Setup

The experimenters designed the workshop as a game for students, held at the students' schools. The objective of the game was for students to become more aware of physical mechanisms around them. The students were to move around the schoolyard, photograph objects with moving parts, and annotate the photographed objects to show which parts moved and how they moved. Afterward, students were to share their thoughts regarding the photographs and sketches that they had created.

The experimenters configured the mobile client on a Pocket PC device to capture a photo and annotation. The experimenters also brought a laptop to the schools, primarily to monitor captured events in real-time (with the help of the desktop platform). The experimenters also ran a Momento server on the laptop and configured the laptop for peer-to-peer wireless networking. In this way, the mobile devices could connect directly to the server running on the laptop

without depending on an external network (there was none at the schools).

To run the workshops, experimenters divided the participants into small groups and showed each group how to use the mobile application. They then let the students take and annotate photos. During this phase, the experimenters and the students' teacher monitored the groups to help them recover from mistakes (most commonly, accidentally exiting the application). Afterward, the experimenters reviewed captured data with the students in a classroom setting.

Participant experience

Although this experiment required participants to use a new device, it avoided some issues, such as carrying an extra device, by being constrained to a workshop setting. In interviews, experimenters reported that participants had little to say about the mobile application itself, but it was clear that with only a brief training session they were able to use the application to take and annotate photos. The only problem reported was that occasionally one group's device launched a different application that momentarily distracted from the main task. The experimenters also noted that participants universally liked using the application.

The teachers' experiences were also largely positive. Based on her experience with PhotoSketch, one teacher is integrating a series of similar workshops into her curriculum.

Experimenter experience

Experimenters felt that using Momento was a large improvement over their fall back approach, having students manually draw both the object and the motion of the object. Because of the work involved in coordinating with schools and teachers and developing and iterating the overall workshop design, experimenters had little time to develop a mobile prototype, but were able to do so with Momento in minutes. One experimenter said that the use of a mobile application "simply wouldn't have happened" without Momento.

Experimenters wanted slightly more control configuring the mobile client than was supported at the time. For example, they asked us to add support for a color scheme and lengthier text description. Also, for this application experimenters needed to configure the desktop platform, server, and client to work on an ad hoc network. To support this, no adjustments were necessary to the desktop platform nor mobile client. However, the server configuration had involved some complicated manual processes that the experimenters initially found too difficult to complete. To address this, we streamlined server configuration to the point that the experimenters needed only to run four commands to configure the core system and their study.

Discussion

Overall, our studies have shown that Momento can be a powerful tool for ubicomp experimenters and a usable tool for participants. Although the studies just described provided overall validation for our concept, they also provided valuable feedback about the features and structure of Momento, which was a basis for iterative improvement.

In addition to the three experiments described in depth here, we ran multiple pilots and one extended study (Scribe4Me, described earlier in the paper). All of these experiences also contributed to Momento's iteration. Some of the important features of Momento that were influenced or identified during iteration include:

- Increased support for piggybacking on existing devices (AwarenessBoard)
- Addition of support for time and place (AwarenessBoard)
- Expanded and more sophisticated networking support (AwarenessBoard)
- Better privacy support when viewing data on the desktop platform (Informal Diary)
- Better support for sifting through hundreds of captures, an opposite problem to data sparsity (Informal Diary)
- Privacy-sensitive web access (Informal Diary)

RELATED WORK

Situated experimentation is an area receiving much attention in ubicomp and mobile device research. Laboratory studies can uncover some interface and navigation flaws [3,20], especially when the physical configuration of the study embodies some aspects of a field setting [21]. However, as Zhang *et al.*'s review of emerging mobile and ubicomp research trends shows, the difficulty of matching the realism of rich mobile contexts in laboratory settings makes field studies paramount because they can unearth unexpected behaviors and adaptations [28]. For example, Benford *et al.* coordinated one of the most extensive field deployments of a ubicomp technology, Can You See Me Now?, a mobile game in which participants raced through city streets to catch virtual avatars that they tracked on mobile devices [2]. The authors compiled data from a variety of sources, including videos, interaction logs, voice and text communications between players, and interviews, and found that connectivity and location tracking irregularities could be co-opted and integrated into the game as a *feature* rather than an error. Halloran *et al.* also showed that long term, situated design can promote a sense of ownership of the technology amongst stakeholders, encouraging adoption [12]. However, Zhang *et al.* and Davies *et al.* note that researchers find it difficult to make use of real devices and to observe and collect data in the field, issues that Momento can address [8].

From a tools perspective, many researchers have developed systems that support the rapid creation of throw-away functional prototypes to be used during iterative design (*e.g.*, [10, 13, 22, 23]). These tools may use a mixture of Wizard of Oz, pre-existing library elements, and coding. For example, Topiary can be used to prototype location-enhanced applications in which a wizard typically shadows a participant and enters location information as the participant moves around [23]. DART extends Macromedia Director to allow designers to Wizard of Oz augmented reality applications. It provides multimedia support for video, tracking, and sensor data [10]. iTools integrates support for evaluation into their prototyping platform by showing the design of a task and the user data derived from completing that task together in one interface [22]. A handful of other projects have taken our approach of using SMS and MMS to enable realistic studies (*e.g.*, [1,6,15,16]), and other tools have been developed spe-

cifically to support event-contingent ESM [11, 17, 19]. Also, InterviewViz supports assisted photo elicitation [26]. None of these tools provide integrated, usable support for experimentation to the extent that Momento does. In addition to integrating multiple experimental methods, Momento includes a collection of novel features ranging from privacy sensitive review to live monitoring and feedback that were motivated by our user-centered, iterative approach.

Recent work has suggested guidelines for tools that support experimentation. For example, in an analysis of several field evaluations of ubicomp applications, Crabtree *et al.* elaborated the need for tools that support side-by-side reviews of video and log data; mechanisms to support user captured video; synchronization between recordings; a timeline-based visualization interface for rapid review of different data sources at different times; and better support for annotation [2, 7]. Morrison *et al.* implemented some of these concepts to support participant observation and logging of mobile application use [25].

CONCLUSIONS AND FUTURE WORK

We presented the user-centered design of Momento, a tool that provides integrated support for situated evaluation of ubicomp applications. Momento provides a desktop platform that connects experimenters with participants in the field, and includes a simple, configurable application extending the capabilities of mobile devices to support participant data collection. Our validation demonstrates how Momento was iteratively designed to meet the needs of participants and experimenters in four separate studies that involved a variety of tools and evaluation methods.

Currently, Momento's primary limitation is that it does not support applications that require rapid synchronization or streaming between the client and server (*e.g.*, mobile videoconferencing). In the future, we plan to add support for streaming media as well as explore models for supporting dynamic coordination among multiple experimenters. Also, we are investigating ways to merge into one visualization data gathered using Momento with data gathered using other methods, such as interview transcriptions.

Momento is open-source software and is available at `http://www.m0ment0.com/`.

ACKNOWLEDGMENTS

This work was sponsored in part by Intel and by NSF grants IIS-0205644 and IIS-0511895. We thank Scott Hudson, Anind Dey, Peter Lyman, Tara Matthews, Heather Horst, Dan Perkel, Christo Sims, Jonathan Hey, Jaspal Sandhu, Susan Fussell, and Peter Scupelli for their help.

REFERENCES

1. M. Ananny, K. Biddick, and C. Strohecker. Constructing public discourse with ethnographic/SMS "texts?". In *MobileHCI'03*, 368–373, 2003.
2. S. Benford, A. Crabtree, M. Flintham, A. Drozd, R. Anastasi, M. Paxton, N. Tandavanitj, M. Adams, and J. Row-Farr. Can you see me now? *TOCHI*, **13**(1):100–133, 2006.
3. A. H. Betiol and W. de Abreu Cybis. Usability testing of mobile devices: A comparison of three approaches. In *IFIP INTERACT'05*, 470–481, 2005.
4. S. Carter and J. Mankoff. When participants do the capturing: The role of media in diary studies. In *CHI'05*, 899–908, 2005.
5. S. Carter, J. Mankoff, S. Klemmer, and T. Matthews. Exiting the cleanroom: On ecological validity and ubiquitous computing. *HCI Journal*. To appear.
6. S. Consolvo and M. Walker. Using the experience sampling method to evaluate ubicomp applications. *Pervasive Computing*, **2**(2):24–31, 2003.
7. A. Crabtree, S. Benford, C. Greenhalgh, P. Tennent, M. Chalmers, and B. Brown. Supporting ethnographic studies of ubiquitous computing in the wild. In *DIS'06*, 60–69, 2006.
8. N. Davies, J. Landay, S. Hudson, and A. Schmidt. Rapid prototyping for ubiquitous computing. *Pervasive Computing*, **4**(4):15–17, 2005.
9. A. K. Dey, G. D. Abowd, and D. Salber. A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. *HCI Journal*, **16**(2-4):97–166, 2001.
10. S. Dow, J. Lee, C. Oezbek, B. MacIntyre, J. D. Bolter, and M. Gandy. Wizard of oz interfaces for mixed reality applications. In *Extended abstracts of CHI'05*, 1339–1342, 2005.
11. J. Froehlich, M. Y. Chen, I. E. Smith, and F. Potter. Voting with your feet: An investigative study of the relationship between place visit behavior and preference. In *Ubicomp'06*, 333–350, 2006.
12. J. Halloran, E. Hornecker, G. Fitzpatrick, M. Weal, D. Millard, D. Michaelides, D. Cruickshank, and D. D. Roure. Unfolding understandings: co-designing ubicomp *in situ*, over time. In *DIS'06*, 109–118, 2006.
13. B. Hartmann, S. R. Klemmer, M. Bernstein, L. Abdulla, B. Burr, A. Robinson-Mosher, and J. Gee. Reflective physical prototyping through integrated design, test, and analysis. In *UIST'06*, 299–308, 2006.
14. J. Heer, S. K. Card, and J. A. Landay. prefuse: a toolkit for interactive information visualization. In *CHI'05*, 421–430, 2005.
15. R. M. Hogarth. Is confidence in decisions related to feedback? In K. Fiedler and P. Juslin, editors, *In the beginning there is a sample: Information sampling as a key to understand adaptive cognition*. Cambridge University Press, 2004.
16. S. Hulkko, T. Mattelmäki, K. Virtanen, and T. Keinonen. Mobile probes. In *NordiCHI'04*, 43–51, 2004.
17. G. Iachello, K. Truong, G. Abowd, G. Hayes, and M. Stevens. Event-contingent experience sampling to evaluate ubicomp technology in the real world. In *CHI'06*, 1009–1018, 2006.
18. Infotrends. Mobile Imaging Study. http://www.capv.com/home/Multiclient/mobileimaging2.html, 2006.
19. S. S. Intille, E. M. Tapia, J. Rondoni, J. Beaudin, C. Kukla, S. Agarwal, L. Bao, and K. Larson. Tools for studying behavior and technology in natural settings. In *Ubicomp'03*, 157–174, 2003.
20. A. Kaikkonen, A. Kekäläinen, M. Cankar, T. Kallio, and A. Kankainen. Usability testing of mobile applications: A comparison between laboratory and field testing. *Journal of Usability Studies*, **1**(1):4–17, 2005.
21. J. Kjeldskov, M. B. Skov, B. S. Als, and R. T. Høegh. Is it worth the hassle? Exploring the added value of evaluating the usability of context-aware mobile systems in the field. In *MobileHCI'04*, 61–73, 2004.
22. S. R. Klemmer, A. K. Sinha, J. Chen, J. A. Landay, N. Aboobaker, and A. Wang. SUEDE: A Wizard of Oz prototyping tool for speech user interfaces. In *UIST'00*, 1–10, 2000.
23. Y. Li, J. I. Hong, and J. A. Landay. Topiary: A tool for prototyping location-enhanced applications. In *UIST'04*, 217–226, 2004.
24. T. Matthews, S. Carter, J. Fong, C. Pai, and J. Mankoff. Scribe4me: Evaluating a mobile sound translation tool for the deaf. In *Ubicomp'06*, 159–176, 2006.
25. A. Morrison, P. Tennent, and M. Chalmers. Coordinated visualisation of video and system log data. In *CMV'06*, 91–102, 2006.
26. N. A. Van House. Interview viz: Visualization-assisted photo elicitation. In *Extended abstracts of CHI'06*, 1463–1468, 2006.
27. A. Weilenmann. Negotiating use: Making sense of mobile technology. *PUC*, **5**(2):137–145, 2001.
28. D. Zhang and B. Adipat. Challenges, methodologies, and issues in the usability testing of mobile applications. *IJHCI*, **18**(3):293–308, 2005.