Prototypes in the Wild: Lessons from Three Ubicomp Systems

Evaluations of three ubicomp systems at multiple design stages provide a better understanding of how ubicomp evaluation techniques should evolve.

Scott Carter
University of California, Berkeley

Jennifer Mankoff
Carnegie Mellon University

The study of ubiquitous computing is concerned with enabling a future in which the most useful applications of such technology are feasible to build and pleasing to use. Feasibility depends on the availability of network connectivity and data, of sensors and algorithms for interpreting the data they produce, and of tools with which to ease the building of applications. The degree to which a ubicomp application is pleasing and useful depends on the designer's ability to understand and meet user needs—needs that evaluation techniques can help designers understand.

One of the hardest problems that application developers face is evaluating ubicomp applications, such as those that adjust output on the basis of complicated contextual cues or that use natural input technologies such as gesture and speech. Ubicomp researchers are beginning to explore evaluation techniques including field studies that drive invention, early-stage requirements gathering, and prototype iteration. For example, Sunny Consolvo and her colleagues recently published an evaluation of an enhanced biology lab they evaluated using lag sequential analysis (the first time that technique had been applied to a ubicomp system).[1] Jennifer Mankoff and her colleagues have developed modifications to heuristic evaluation that enhance its applicability to a subdomain of ubicomp, ambient displays.[2]

Evaluation is crucial at all design stages, and the best designs include evaluations that involve users in the design process throughout a series of iterations. Evaluation research can help us develop a suite of techniques applicable to finished systems (such as Consolvo and her colleagues' work) and early stage design (such as Mankoff and her colleagues' work).

Case studies

We built and tested three ubicomp systems and found difficulties that stand in the way of such evaluation. Jennifer Mankoff and Bill Schilit developed the first, PALplates, in 1996.[3] Mankoff and her colleagues designed and evaluated the second, a nutritional tracking system, between 2001 and 2004.[4] Mankoff, Scott Carter, and Patrick Goddi developed and evaluated the third, Hebb—our keystone study—from 2002 through 2004.[5]

PALplates

We created the PALplates project to support office workers in doing everyday tasks by presenting key information and services at *places of need*, or locations where workers were most likely to need them. PALplates were interactive computing platforms, each providing unique services on the basis of its location in the office. Our goal was to create a system that, although only available at certain places, would feel ubiquitous to users. We tried to do this by placing a system at each place users were likely to be doing a task that might require computational support. This could free users from the need to constantly carry a device.

For example, suppose a user goes to the printer room to pick up a print job and notices that the printer is almost out of paper. He or she might use the print room PALplate to request extra paper or resubmit the print job to a different printer. In contrast, an employee might use the meeting room PALplate to reserve that room for a follow-up meeting to one just completed.

Because this was a novel approach to ubiquitous computing, we wanted to get user feedback about such a system's effectiveness as soon as possible. We wanted to know if people would use the devices and whether they used them in a location-specific way. We also wanted to know whether users wanted additional services and whether they found the services in our initial set useful. Our goal was to generate requirements for our next iteration.

It was best to let users experience this novel approach to ubiquitous computational support. Yet implementing even a prototype of our system ubiquitously would have required installing infrastructure in many locations. Instead, we decided to evaluate a paper prototype of the system in the field.

Figure 1. A paper prototype of the PALplates office worker support system.

Evaluation. Researchers traditionally do paper prototyping by sketching all of a GUI's dialog boxes, screens, and other interactive elements on paper. An evaluator then simulates a computer's reactions as users execute predefined tasks using the paper interface.[6] This technique is effective and quick, in part because it requires no coding. You can also use it to involve end users in design. For example, Michael J. Müller presented end users with a kit of user interface elements and asked for feedback about layout.[7]

Because PALplates needed to be available in many places at all times of day for users to perceive them as ubiquitous, we weren't able to apply the technique in its classic form. Instead, we created an interface that users could manipulate even when experimenters weren't present. We posted notebooks decorated with icons around the office (on office doors, in common rooms, and in conference rooms), along with instructions, pens, and Post-it notes that users could interact with. Figure 1 shows an example of these "Paper Plates." Users could write notes or make requests on these interfaces. They could also make suggestions and reservations and see local news.

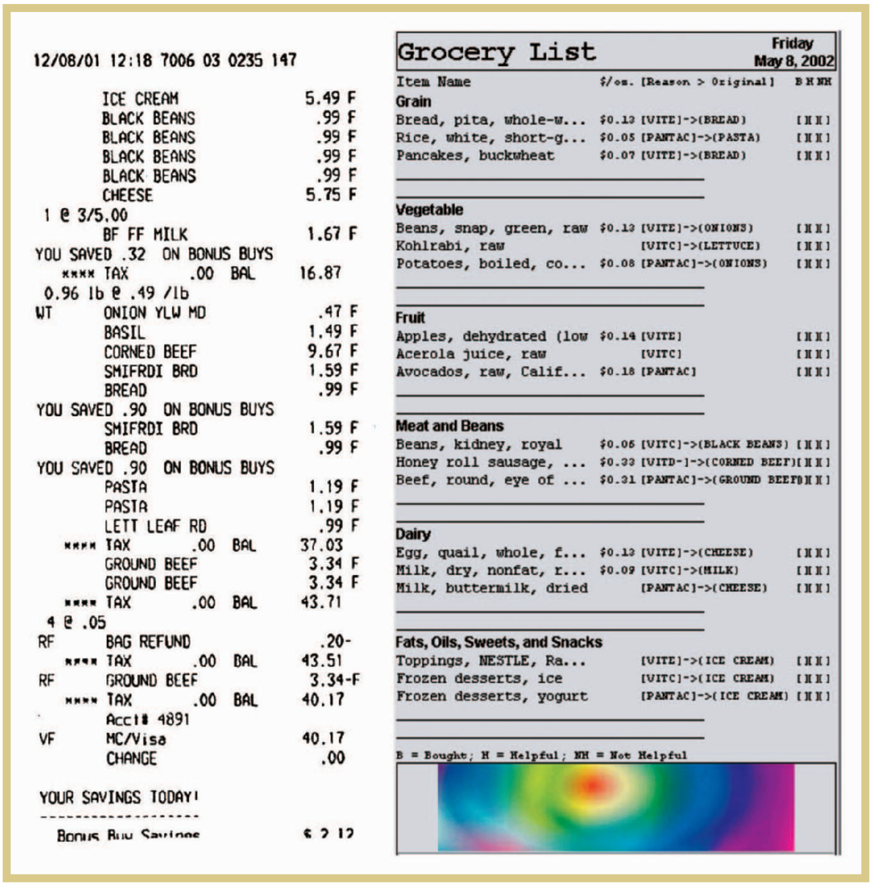
To make the system interactive, rather than updating the interface in real time, we asked five volunteers (friends and coworkers) to play the part of a network that we nicknamed sneakernet by responding to user requests. Sneakernet, which ran once per day, would schedule meetings, fetch supplies, and deliver Post-it notes to other Paper Plates based on user requests. For example, a sneakernet member might pick up a note and deliver it to the display to which it was addressed.

We deployed 16 Paper Plates around our office. Most were on personal office doors, but one was in the copy/printer room, one was in the kitchen (which opens onto a lounge where people can congregate), and one was in a meeting/conference room where we also had our weekly kaffeeklatsch. This covered the most-used public spaces in our small office, as well as the office doors of many researchers and staff. Because anyone could interact with the public displays, we don't know exactly how many people used the Paper Plates, but we believe that most of the 20 or so employees used the system at least once.

Our analysis was mostly qualitative. We monitored how people used Paper Plates in different places. We asked sneakernet members to report interesting uses or comments. And we interviewed users to ask them about their experiences using the system and what they would like to see changed.

We found that even though the Paper Plates were missing important features that would be present in a full-fledged application, people used them, and for different tasks in different locations. For example, they used the display in the kitchen mostly for discussion and ordering kitchen supplies. Although we expected functions to be closely tied to location, we also found "remote access" activity. For example, people sometimes reserved the meeting room from a Paper Plate located in the hallway. In addition, people were interested in using the PALplates to access their own private information such as Web documents and calendars.

Summary. Overall, paper prototyping worked for us in this case. It allowed us to explore whether providing computing services at points of need was viable. It also helped us test four services and generate new requirements for additional services that such a system would need to support.



12/08/01 12:18 7006 03 0235 147
ICE CREAM 5.49 F
BLACK BEANS .99 F
BLACK BEANS .99 F
BLACK BEANS .99 F
BLACK BEANS .99 F
CHEESE 5.75 F
1 @ 3/5.00
BF FF MILK 1.67 F
YOU SAVED .32 ON BONUS BUYS
**** TAX .00 BAL 16.87
0.96 lb @ .49 /lb
WT ONION YLW MD .47 F
BASIL 1.49 F
CORNED BEEF 9.67 F
SMIFRDI BRD 1.59 F
BREAD .99 F
YOU SAVED .90 ON BONUS BUYS
SMIFRDI BRD 1.59 F
BREAD .99 F
YOU SAVED .90 ON BONUS BUYS
PASTA 1.19 F
PASTA 1.19 F
LETT LEAF RD .99 F
**** TAX .00 BAL 37.03
GROUND BEEF 3.34 F
GROUND BEEF 3.34 F
**** TAX .00 BAL 43.71
4 @ .05
RF BAG REFUND .20-
**** TAX .00 BAL 43.51
RF GROUND BEEF 3.34-F
**** TAX .00 BAL 40.17
Acct# 4891
VF MC/Visa 40.17
CHANGE .00
YOUR SAVINGS TODAY!

Bonus Buy Savings $ 2.12

Grocery List
Friday
May 8, 2002
Item Name $/oz. [Reason > Original] B H NH
Grain
Bread, pita, whole-w... $0.13 [VITE]->(BREAD) [ ][ ][ ]
Rice, white, short-g... $0.05 [PANTAC]->(PASTA) [ ][ ][ ]
Pancakes, buckwheat $0.07 [VITE]->(BREAD) [ ][ ][ ]
Vegetable
Beans, snap, green, raw $0.13 [VITE]->(ONIONS) [ ][ ][ ]
Kohlrabi, raw [VITC]->(LETTUCE) [ ][ ][ ]
Potatoes, boiled, co... $0.08 [PANTAC]->(ONIONS) [ ][ ][ ]
Fruit
Apples, dehydrated (low $0.14 [VITE] [ ][ ][ ]
Acerola juice, raw [VITC] [ ][ ][ ]
Avocados, raw, Calif... $0.18 [PANTAC] [ ][ ][ ]
Meat and Beans
Beans, kidney, royal $0.06 [VITC]->(BLACK BEANS) [ ][ ][ ]
Honey roll sausage, ... $0.33 [VITD-]->(CORNED BEEF)[ ][ ][ ]
Beef, round, eye of ... $0.31 [PANTAC]->(GROUND BEEF)[ ][ ][ ]
Dairy
Egg, quail, whole, f... $0.13 [VITE]->(CHEESE) [ ][ ][ ]
Milk, dry, nonfat, r... $0.09 [VITC]->(MILK) [ ][ ][ ]
Milk, buttermilk, dried [PANTAC]->(CHEESE) [ ][ ][ ]
Fats, Oils, Sweets, and Snacks
Toppings, NESTLE, Ra... [VITE]->(ICE CREAM) [ ][ ][ ]
Frozen desserts, ice [VITC]->(ICE CREAM) [ ][ ][ ]
Frozen desserts, yogurt [PANTAC]->(ICE CREAM) [ ][ ][ ]
B = Bought; H = Helpful; NH = Not Helpful

Figure 2. (a) A receipt from a local grocery store and (b) a shopping list our system generated based on that receipt.

Nutrition tracking

Figure 2 shows an application we built that uses inexpensive, low-impact sensing to collect data about what household members are purchasing and consuming and then uses simple yet persuasive techniques to suggest potential changes. Healthy eating can help to reduce obesity and, consequently, the chance of developing chronic diseases such as diabetes.[8] Yet many people don't know how many servings of fruits, grains, vegetables, and fats they're eating or which nutrients are missing in their diet.

The proposed system gathered data about purchasing habits when a user scanned in receipts with a handheld scanner (for example, during bill sorting at the end of the week). A shopping list, printed at the user's request, provided annotated suggestions for slight changes in purchases. This piece of paper provided suggestions for a more balanced diet (based on USDA guidelines) at the most pertinent moment—when the user was making purchasing decisions. It could encourage healthier purchases such as baked tortillas instead of chips or whole-wheat bread instead of white. The system also displayed a food pyramid skewed to indicate the relative amounts of different foods a user was purchasing, enabling him or her to easily identify areas needing change.

As with PALplates, our goal was to provide a computational service without requiring the user to constantly carry a computer. Again, we wanted to know if people would use our device. Was the printed shopping list sufficient to meet a person's needs while shopping, and were our recommendations useful? We also questioned whether the process of using our device fit well into a person's everyday shopping patterns. Finally, we questioned whether people could easily interpret the information the food pyramid displayed. Answering these questions was challenging. Because the system needed to integrate into a person's shopping pattern, we had to deploy a working prototype in a field setting—a home. Additionally, our recommendation system required historical information and used a complex algorithm. This wasn't easy to simulate. Despite this, we wanted to test as many aspects of the system as possible before we put too much effort into developing a prototype.

Evaluation. We began our evaluation by surveying 57 shoppers at a local supermarket and interviewing five people to guide our design's early stages. In our interviews, we began by asking people about their shopping habits, their use of shopping lists, and so on and demonstrated paper prototypes of our planned interface. For our qualitative analysis, we looked for patterns in our participants' shopping habits that could guide system placement (people preferred the kitchen), our design for motivating change (discounted prices were the strongest motivator), and other system features. On the basis of our results, we added the capability to base recommendations on dynamically downloaded sales data.

We then deployed the system to three users (ages 23 to 38). Our participants lived alone or with one other adult, shopped frequently (weekly to daily), ate out infrequently, and used shopping lists. We interviewed the participants and asked them to complete home inventories and send us receipts for two weeks before deployment began to build up a historical picture of their purchasing habits. We then gave each participant a system to use for three weeks. During those three weeks, they completed weekly surveys. The deployment ended with another interview.

Participants felt that the process of using the system was straightforward,

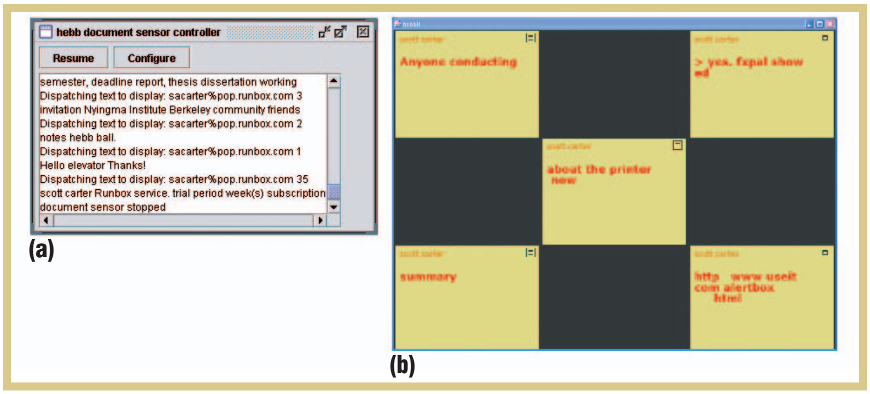


Figure 3. Hebb (a) interest sensor and (b) public display. The interest sensor scans participants' email for topics of interest. The public display collects topics and displays those likely to encourage communication and collaboration between groups.

but they were frustrated by its instability. However, we learned that our recommendation algorithm had serious flaws: Our suggestions included foods disliked by or unfamiliar to participants and didn't appear to vary much with each new shopping trip. Also, the modified food pyramid was too hard to understand, and users didn't feel it reflected their diets. The problems uncovered during deployment weren't particularly surprising for a new system's first iteration, but the study's development effort and high overhead were totally out of proportion.

We decided to address one issue, the food pyramid problems, using paper prototyping. We created several iterations of our modified food pyramid in an attempt to create something that was intuitive and easy to read at a glance, with a low learning curve. This let us successfully address the specific issue of readability.

We conducted a second deployment after updating our algorithms and attempting to increase our system's robustness. This study was structured like the previous deployment, except that we extended the system use phase from three weeks to five to address a complaint that three weeks wasn't enough time. Also, we used log files combined with weekly reports our subjects filed to construct a timeline of system use. This became the basis for our final interview, allowing us to more easily gather specific data about use experience. This study's three participants were 27 to 33 years old and single, shopped for groceries at least once per week, and ate out rarely (less than once a month).

Our analysis of the final study was limited by the fact that, once again, instabilities in our technology caused daunting problems. Despite a significant development effort over three years by a group of five students, the system still had major bugs. At the end of five weeks, only one of our systems was still up and running. Luckily, when the systems worked, we got rich feedback. In fact, the feedback quality suggests that simulation, even with its unavoidable gaps in service, might have been just as effective at testing our ideas.

We found that participants were motivated by the wish to see their purchases' nutritional balance reflected in the food pyramid. We also found that they viewed the shopping list more as a source of ideas than as a tool to use while shopping. For example, one participant remarked, "I just reviewed them and kept the suggestions in mind." Along those lines, participants were very quick to either toss away or pick up suggestions. They ignored bad suggestions while considering intriguing ones (such as those diversifying their diets). Our attempts to augment the shopping process failed because our printout didn't exactly fit existing habits.

Summary. In retrospect, we had to spend a lot of time and effort on this system before learning which aspects were useful and which weren't. The initial interviews we conducted had limited value when compared to what we learned when we actually deployed the system. The paper prototyping was very successful but only let us answer a narrowly defined question. The deployments were highly informative despite being crippled by problems with placing an early prototype in the field. However, the deployment cost was unduly high. Simulation appears to be a viable alternative to consider.

Hebb

It's not uncommon for multiple small working groups in a large organization to overlap in interests, hobbies, or the problems they're trying to solve. Yet often, such shared interests go undiscovered even among loosely coupled groups that share common spaces or are located in the same building. Better awareness could lead to fruitful collaborations, friendships, and other positive social change. We designed our Hebb system to capture and convey shared interests. This system arose from a series of formative interviews that we conducted with members of six small working groups. From these interviews, we found that the benefits of collocation (awareness of shared interests) often don't extend beyond the group.

In response to this issue, we designed a system that senses group members' interests via email analysis software and displays relationships between members on public and private displays to encourage conversation about those topics. The Hebb system includes interest sensors, presence sensors, and public and private displays (see figure 3).

The interest sensor generates name and keyword event pairs, and the system generates keywords from user email. The interest sensor also makes available encrypted full document data for use on personal PDAs. The presence sensor generates unique user identifiers for each user sensed in the space via either RFID badging or the presence of the user's PDA on the local wireless network. The public display generates events indicating from which document keywords were recently displayed. Servers on each component and a remote discovery server handle subscription and networking.

We built Hebb because we wanted to know if it was possible to help loosely coupled small groups build a better awareness of cross-group shared interests. Although we could test portions of the system in the lab, the only way to know if it really changed group dynamics was to deploy it in the field over a long period of time. Additionally, the system's complexity and the interaction of multiple devices across time and space and within the context of other activities made it next to impossible to design a lab study that would give us useful feedback.

Evaluation. Although our early work with Hebb included focused paper prototype work (similar to that in the nutrition study) and interviews, our evaluation's most important component was a series of field studies. After attempting to recruit several different groups to use the system, we deployed it with two research groups in the same department.

Our first attempts to recruit groups to use the system met with unexpected resistance. We gave multiple presentations for multiple work groups but found it difficult both to find support for the system and to convey an appropriate conceptual model of the system. Some groups rejected the system outright because they felt it would increase email spam and that the public displays would end up showing random ads. Further interviews with these groups showed that these responses seemed to reflect an accepted ideology about technology, combined with little previous experience with novel technologies. Thus, we faced a chicken-and-egg problem: Convincing someone who hadn't used a novel technology to use Hebb was difficult, even though our participants quickly accepted it once we deployed it. We scaled down our hopes of deploying the system to a group that was unbiased with respect to ubiquitous computing and eventually settled on two academic research groups. Interviews revealed that these groups shared a considerable amount of work practice experiences but didn't communicate with one another. Thus, the two groups seemed a good match for the Hebb system.

Even then, we ran into problems regarding technology acceptance. One group readily accepted the system. However, we found it difficult to convince the other that the system was secure. Also, some users vehemently objected to the notion of an email sensor. Again, explanations of the system's privacy assurances—that the system would transmit only high-level data and that they could easily turn off the sensor—did little to placate these users.

After negotiations with this group stalled, we deployed the system to two other academic research groups working on similar topics. The groups were spread across three different spaces: One group of three members was collocated, while the other group of four was split between two spaces. This particular arrangement allowed us to determine how well the system supports intra- and intergroup awareness and communication. We deployed the interest sensor first for four weeks to establish common email patterns and then deployed the rest of the system for another four weeks.

During the deployment, we logged all user interactions with the public and private display. We also monitored and interviewed user groups to determine use that didn't appear in logs. We were particularly interested in ascertaining the extent to which participants used the public display to maintain peripheral awareness of other users' interests. We also cataloged situations in which users verbally discussed something that they noticed on the display without interacting with the system in any way.

Data analysis showed that the system helped increase awareness and communication among study participants. In particular, we found that users tended to email each other more regularly after the deployment than before. Also, we found that users' attitudes toward the interest sensor's implicit sensing evolved from strongly against it at the beginning of the deployment to accepting toward the end. Another finding was that users showed a strong desire to have the peripheral display separate from their work display. We also discovered that in peripheral displays, visual onset (for example, objects appearing and disappearing but never moving) captured user attention without causing annoyance.

Summary. Overall, we found deploying this system challenging because participants had difficulty understanding the model of interaction and because discovering errors and updating the systems in the field was difficult and time consuming. However, we found that users began adapting to the system rapidly and derived value from it.

Implications for evaluating ubicomp applications

In our case studies, we struggled to balance quality of evaluation and ease of prototyping. We found interactivity to be the most important aspect of a system for this balance.

Noninteractive prototypes

We used noninteractive prototypes in all three projects. In the PALplates project, we used prototypes in field settings. Paper prototyping was very effective for several things we needed to test. Most importantly, it enabled us to quickly deploy enough prototypes to give users a sense of ubiquity and to get solid feedback on our idea of point-of-need computing. Additionally, paper was an excellent medium for supporting a discussion forum that was place based.

Successes and limitations. However, paper prototyping has serious limitations. In particular, as it stands, paper prototyping fails to adequately handle scale (Linchuan Liu and colleagues arrived at a similar finding[9]). For example, in PALplates, we could only update our displays once daily. We were unable to simulate finer-grained control such as updating the printer status each time a job began or ended or updating a map showing the location of everyone in the building. Also, with human agents, demonstrating realistic errors was difficult. Instead, things were perhaps too perfect. Given the number of volunteers needed to run the system, anything more sophisticated would have required too much training. Also, participants knew humans were involved and expected fairly accurate, consistent behavior.

In our nutrition tracking study, we modified our food pyramid in an attempt to create something that was intuitive and easy to read. Our approach to this was static and our results were limited. However, we were able to address the specific issue of readability.

Our nutrition tracking paper prototype couldn't help us assess our recommendations' quality. They depended on weeks of data that we couldn't gather and integrate without a working system, which would have been much more time consuming to construct. This data was particularly difficult to acquire because the system had to be embedded in participants' regular shopping routine.

In our Hebb study, we evaluated paper prototypes of the public display. Although we gained some feedback on the graphics used in the display itself, we found that we were unable to evaluate the display's peripherality. Because the display lacked interactivity, we had to update it by hand, making it difficult to monitor the display while focusing on some other task.

Conclusions. In sum, we determined that paper prototyping is useful for rapid iteration of graphic designs in lab settings and for getting early feedback on an application's situated use. However, paper prototyping isn't as useful for testing interactivity or peripherality in lab settings. Also, paper prototyping can't be scaled and generalized, is problematic for synchronous activities, and is a difficult medium for testing error and ambiguity.

Interactive prototypes

We used field experiments with interactive prototypes in the nutrition tracking and Hebb projects. The biggest lesson we learned from the nutrition tracking project was the difficulty of accurately assessing the problems and potential of a system that must integrate unobtrusively into a daily life activity. Our early interviews failed to alert us that our system wouldn't integrate well into shopping patterns. Also, the field experiment helped indicate what level of ambiguity was acceptable to users.

Successes and limitations. In the Hebb deployment, we again found unobtrusiveness difficult. In particular, minimizing updates to maintain the system's unobtrusiveness is important. Upon deploying the system, several unanticipated needs and bugs arose, and we had to fix software and hardware in situ. One way to mitigate these reinstallations' effect is to use public and remotely accessible components rather than purely private ones. In many cases, such as cellular phone deployments, installing private applications is necessary, but they should be as lightweight as feasible. Users would much rather be told that there was a change in the system than have to make the change themselves.

Another approach that we found useful to mitigating the system's obtrusiveness was recruiting a local champion. Such a person must be a member of the deployment group, knowledgeable about the group, and capable of selling the technology to administrators and directors while eliciting grassroots support for the technology's adoption. In our deployment, the only successful installation outside our lab depended on such a person. Past work has shown local champions' presence to be important in integrating a technology with existing patterns of work and to encourage uptake.[10]

However, we found that users' attitudes toward the system changed as the system became less remarkable and more unobtrusive. The use of all applications evolves with time, but this evolution is exacerbated in ubicomp applications because they represent a new interaction paradigm for users. In our study, participants tended to adapt their use and perception of peripheral output technologies over the course of deployment. While at first they found it difficult to use common technologies in new ways, they eventually adapted and felt comfortable with these new use modes. This finding resonates with other work in which "adaptation, consideration, and time" were necessary for ubicomp technologies to succeed in everyday situations.[11]

We also found a shift in user attitudes with regards to privacy. Privacy is a major concern for any centralized ubicomp system because a plethora of user data could be collected without the user's explicit consent. We found that the balance between privacy and usefulness was always in flux during our deployment. At first, we thought that because the public displays showed limited information, users would have few privacy concerns. In fact, they were so concerned with the interest sensor's data mining that the public display was unimportant.

Finally, we found that users struggled to develop a conceptual model of the system. Moreover, early attempts to deploy the system revealed that users felt that they were involved in an experiment that didn't hold any benefits for them; they felt as though they were guinea pigs testing an already-developed system. A better model might be to involve these users in the iterative design process's early stages.

Conclusions. Based on our experiences, we feel that field-based interactive prototypes provide invaluable feedback on a system's use and co-evolution. However, they're difficult and time consuming to deploy, and maintaining them unobtrusively is challenging. Designing for remote updates and using local champions and participatory design might mitigate these issues.

As we mentioned earlier, testing ubicomp applications in field settings is particularly important. And although the approaches we discussed here can help, it's still difficult to field test and maintain a fully implemented system. Thus, a need exists for tools that allow researchers to rapidly deploy ubicomp applications in field settings. These tools should have paper prototyping's positive characteristics (ease of creation and deployment) but should also be scalable and interactive. While some tools are emerging for rapid field testing,[12,13] much work remains to be done.

The lessons we learned by evaluating ubicomp systems at different stages of design should be beneficial to researchers requiring insights into issues such as when to expose users to different prototype fidelities and the range of costs and benefits associated with deploying prototype systems. We hope our work encourages researchers to continue to bridge the gap between ease of iteration and experimental realism.

REFERENCES

1. S. Consolvo, L. Arnstein, and B.R. Franza, "User Study Techniques in the Design and Evaluation of a Ubicomp Environment," *Proc. 4th Int'l Conf. Ubiquitous Computing* (UbiComp 2002), LNCS 2498, G. Borriello and L.E. Holmquist, eds., Springer, 2002, pp. 73–90.
2. J. Mankoff et al., "Heuristic Evaluation of Ambient Displays," *Proc. SIGCHI Conf. Human Factors in Computing Systems* (CHI 03), ACM Press, 2003, pp. 169–176.
3. J. Mankoff and B. Schilit, "Supporting Knowledge Workers Beyond the Desktop with PALplates," *Proc. SIGCHI Conf. Human Factors in Computing Systems* (CHI 97), ACM Press, 1997, pp. 550–551.
4. J. Mankoff et al., "Using Low-Cost Sensing to Support Nutritional Awareness," *Proc. 4th Int'l Conf. Ubiquitous Computing* (UbiComp 2002), LNCS 2498, G. Borriello and L.E. Holmquist, eds., Springer, 2002, pp. 371–378.
5. S. Carter, J. Mankoff, and P. Goddi, "Building Connections among Loosely Coupled Groups: Hebb's Rule at Work," *J. Computer Supported Cooperative Work*, vol. 13, nos. 3–4, 2004, pp. 305–327.
6. M. Rettig, "Practical Programmer: Prototyping for Tiny Fingers," *Comm. ACM*, vol. 37, no. 4, 1994, pp. 21–27.
7. M.J. Müller, "Retrospective on a Year of Participatory Design using the PICTIVE Technique," *Proc. SIGCHI Conf. Human Factors in Computing Systems* (CHI 92), ACM Press, 1992, pp. 455–462.
8. S. Hankinson et al., eds., *Healthy Women, Healthy Lives: A Guide to Preventing Disease*, Simon & Schuster, 2001.
9. L. Liu and P. Khooshabeh, "Paper or Interactive? A Study of Prototyping Techniques for Ubiquitous Computing Environments," *Proc. SIGCHI Conf. Human Factors in Computing Systems* (CHI 03), ACM Press, 2003, pp. 1030–1031.
10. E. Rogers, *Diffusion of Innovations*, Free Press, 2003.
11. D. Fitton and K. Cheverst, "Experiences Managing and Maintaining a Collection of Interactive Office Door Displays," *Proc. 1st European Symp. Ambient Intelligence* (EUSAI 03), LNCS 2875, E. Aarts et al., eds., Springer, 2003, pp. 394–409.
12. A.K. Sinha and J.A. Landay, "Capturing User Tests in a Multimodal, Multidevice Informal Prototyping Tool," *Proc. 5th Int'l Conf. Multimodal Interfaces* (ICMI 03), ACM Press, 2003, pp. 117–124.
13. Y. Li, J.I. Hong, and J.A. Landay, "Topiary: A Tool for Prototyping Location-Enhanced Applications," *Proc. 17th Ann. ACM Symp. User Interface Software and Technology* (UIST 04), ACM Press, 2004, pp. 217–226.

the AUTHORS

Scott Carter is a graduate student at the University of California, Berkeley. His research interests are in tools and techniques for designing and evaluating mobile and ubiquitous computing applications, tools for disabled users, computer-mediated communication studies, sports innovation, and visualizations of brain-activation data. He has developed and deployed several ubicomp applications and has contributed to a toolkit that supports the development of peripheral displays. He received his master's degree from UC Berkeley. Contact him at the Human Computer Interaction Inst., School of Computer Science, Carnegie Mellon Univ., 5000 Forbes Ave., Pittsburgh, PA 15213-3891; sacarter@cs.berkeley.edu.

Jennifer Mankoff is an assistant professor in the Human Computer Interaction Institute at Carnegie Mellon University. Her research interests are in tools and techniques for developing rapid, iterative prototypes of ubiquitous computing applications and accessible technology and mediation of ambiguous, recognition-based interfaces. Her work's application areas include assistive technology for people with disabilities and health and safety. She received her PhD in computer science at the Georgia Institute of Technology. Contact her at the Human Computer Interaction Inst., School of Computer Science, Carnegie Mellon Univ., 5000 Forbes Ave, Pittsburgh, PA 15213-3891; jmankoff@cs.cmu.edu.