Rapid Prototyping and User-centred Design of Interactive Display-based Systems

Dan Fitton, Keith Cheverst, Chris Kray, Alan Dix, Mark Rouncefield and George Saslis-Lagoudakis
{df,kc,kray,dix,rouncefield,saslisla} @comp.lancs.ac.uk
Department of Computing,
InfoLab 21,
Lancaster University, Lancaster, UK

Abstract

Rapid prototyping is a broad field encompassing many domains and varied approaches. This article presents our experiences using user-centred rapid prototyping approaches for developing display-based ubiquitous systems, to be deployed over the longer term, in 'real world' situations. In contrast to the usual use of rapid prototyping to produce a single 'proof of concept' demonstrator to investigate technical feasibility, we have found that without investigation of real use, technical feasibility can be meaningless. We have adopted multi-phased prototyping approaches, using different techniques, at different stages, in order to gather feedback. We hope that our findings will help other researchers and practitioners to select appropriate prototyping techniques for future projects.

1. Introduction

Rapid prototyping is an approach used in many domains and one that appears highly suited to research in the area of ubiquitous computing [1]. This approach primarily comes from manufacturing/engineering industries, where the process is the relatively fast physical fabrication of a design or concept for purposes such as demonstration, evaluation, or testing. Rapid prototyping is also used in software engineering, where requirements are uncovered through feedback from analysis of prototypes provided early in the development process. When developing ubiquitous computing systems, which often include both hardware, software and (of course) human factors, these two processes form an ideal design and development methodology.

The goal of much work on display-based ubiquitous systems (and rapid prototyping in general) appears to be more motivated by the production of proof of concept demonstrators, usually in order to gauge technical feasibility and limited initial user feedback, for example [2]. However, in our work we have found that often it is equally important to investigate factors such as use and appropriation [3] and that in some cases, without user studies, technical feasibility can be meaningless.

In this article we explore our use of rapid prototyping together with a phased and iterative user-centred design approach, to develop display-based systems for use over the longer term in 'real world' situations. First we present a brief overview of five prototyped display-based 'ubicomp' systems, in terms of their aims, development approach and development strategy. Following this, we discuss the main issues encountered when applying our rapid prototyping techniques during the development of the five systems. Finally we summarise the lessons learned and present the advantages and disadvantages of the different rapid prototyping techniques used.

2. Prototyped Systems

This section presents an overview of the 'ubicomp' systems that we have developed and deployed using 'rapid' prototyping techniques and a user-centred design approach.

2.1 Hermes 1 – Office Door Display System

Hermes [4] is a system of interactive office door displays (shown in figure 1 top) which provide asynchronous messaging facilities. They are designed to augment (rather than replace) existing messaging practises, such as the use of Post-it™ notes.

2.1.1 Aims

One of the key motivations for developing the Hermes displays was to explore issues of adoption and 'situated interaction' in a location with both public and private elements (i.e. outside an office door). Our aim was to evaluate the system 'in place', allowing it to evolve as further requirements emerged over a significant period of time with daily use from a range of 'real world' users.

2.1.2 Approach

We used prototyping and early deployment including only a small set of features (from those technically possible) thus providing users with an information appliance performing a small number of functions well. The reduced functionality allowed us to concentrate on ease of use and reliability, and the relatively low complexity meant that prototypes could be developed and refined quickly (in the order of days or weeks). From an early stage deployed prototypes included logging of use, the benefits provided by this are discussed later in section 4.

A phased development approach was employed [4], where each phase had a different primary objective and involved several prototyping iterations. The objective of the first phase was to develop core functionality and involved the deployment of door displays outside the offices of two of the system's developers. This ensured some initial testing outside of the 'lab', but with users where low reliability and initial problems were not likely to affect future use. From this initial phase a major issue impacting reliability was uncovered, the fact that humans block wireless network signal when standing in front of a Hermes Display.

Phase two aimed to increase system reliability, solving problems encountered during phase one and three additional door displays were deployed.

The objective of phase three was to add new interaction methods (in response to user feedback), and increase the number of deployed door displays from five to ten. One interesting trade-off which emerged was that users were prepared to forgo security almost completely in order to reduce interaction time.

In order to avoid including only 'techies' we deployed door displays for two departmental secretaries and a sociologist. Later phases involved the addition of new features and further improvements to reliability. Deployment, evolution and use of the prototypes lasted nearly three years, only ending due to a move to a new computing building.

2.1.3 Summary of Development Strategy

Initial feasibility study and requirements gathering through deployment of early prototypes for 'friendly' users, after which deployment was expanded to a range of users. Development process split into phases allowing changes in the development focus, each phase lasting approximately 4 months. Phases split into prototyping cycles, each lasting in the order of days or weeks. User-centred design combined with rapid prototyping to encourage adoption and appropriation by maximising usability. Feedback from users collected using both formal (questionnaires and interviews) and informal (email, complains, chance conversations) means, all user interactions logged.

2.2 Hermes 2 – Redesign and Redeployment of Hermes 1

Following the success of the initial Hermes and SPAM [5] projects, our next aim was to understand the way in which the physical placement and design of networked displays in semi-wild settings influences and facilitates collaboration and community. One of the initial ways we plan to achieve this is by redesigning the original Hermes system, and improving underlying technology. We currently plan to deploy 40 door displays in the new computing department at Lancaster.

Figure 1 – Hermes 1 Deployment (top), a Hermes 2 'Showroom' Configuration (middle) and a Hermes 2 Probe Pack (bottom)

2.2.1 Aims

In Hermes 1 a 'one size fits all' approach was used for door-display configuration (figure 1 top); with the forthcoming redeployment we have the opportunity to investigate what users find acceptable in terms of parameters such as display size, number of displays, housing, supporting infrastructure and user interface layout. Consequently, at this early stage we have two initial aims: to determine the physical form factor and configuration of displays desired by users.

2.2.2 Approach

To investigate our first aim we designed six different hardware and user-interface configurations for door displays, including a range of different sizes and types of displays. These configurations were then rapidly prototyped using appropriate hardware and software, then mounted and arranged to give an impression of how they might appear when deployed (see figure 1 middle), providing showcase scenarios for all six potential door-display configurations together at the same time. We then gave ten future door-display owners a semi-structured tour around the showcases, which were video-taped. Each configuration was explained in the context of a scenario to highlight potential use. The experiment seemed to engage the participants, and having different prototypes next to each other enabled users to pick out the configurations they preferred, a typical comment being 'I like that [user interface] but I'd want it on that display'.

To investigate our second initial aim we have given probe packs [6] to all of the participants in the showcase experiment. These packs contain a diary, instant camera, pen and glue (figure 1 bottom). The diary is intended to record messaging activities over a period of 7 days, with pages for each day to record and describe messages left for others, for oneself and messages left by others. The camera allows participants to takes pictures of messages, which can then be glued into the diary and annotated appropriately to describe its context. Additionally we have carried out some paper prototyping of user interface configurations with limited success, but not as extensively as with systems such as [7].

2.2.3 Summary of Development Strategy

Rapid prototyping of multiple potential hardware and UI configurations for demonstration together in showcases. Potential users given guided tour, in order to choose which configuration they would prefer deployed at their office. Probe packs given to participants in the showcase experiment in order to uncover requirements and provide materials for a forthcoming design workshop.

2.3 Hermes Photo Display

The Hermes Photo Display [8] is an extension of functionality provided in Hermes 1 enabling context sharing using pictures. It enables users to send pictures both locally and remotely (using methods such as MMS, e-mail and Bluetooth) to a location, where they are organised into a presentation and appear on an appropriate display (figure 2). We are attempting to support messaging to a place rather than a person in order to promote a sense of community.

2.3.1 Aims

Our goal is to explore issues of use and user acceptance with this type of shared picture messaging display. To achieve this we are investigating areas such as interaction methods and presentation of information.

2.3.2 Approach

To date our approach has involved rapid prototyping and deployment of a single display, focussing on methods of asynchronous and synchronous interaction through mobile phones. The display was initially deployed for use by a group of 'friendly' users. Once reliability reached an acceptable level the system was redeployed for 'real world' users (figure 2). In order to elicit feedback from users in this public setting, we advertised for volunteers who were then asked to carry out prescribed tasks with the display and then fill in a questionnaire.

2.3.3 Summary of Development Strategy

Investigating the feasibility of a public picture messaging display, in terms of technology supporting the presentation of information and interaction methods. Rapid prototyping using 'off the shelf' and custom hardware and software in order to assess feasibility and enable deployment. Investigation of use through logging and questionnaire-based user study.

Figure 2 – Hermes Photo Display Deployment and User Interface

2.4 SPAM – SMS Messaging Display

The SPAM system provides a lightweight alternative to the usual workplace communication methods (telephone, fax and e-mail). The system enables asynchronous SMS messaging between remote SPAM displays and staff members with mobile phones 'off site'.

2.4.1 Aims

The first aim was to develop and deploy a reliable and easy-to-use communication system based on SMS to enable messaging to a place (in this case an office for staff providing support to former psychiatric patients) rather than a person. The second was for this system to act as a technology probe to help understand this domain further, making logging of user interaction for later analysis crucial [5].

2.4.2 Approach

The initial approach involved a Participatory Design workshop [9] where chosen scenarios (informed by previous ethnographic work) and props (including a Hermes display) were used to generate discussions about requirements and possible technology solutions.

Following the workshop a system was rapidly prototyped based around these requirements using mainly 'off the shelf' hardware and software. This enabled a prototype solution to be produced quickly (in approximately one month), time was then spent on testing and burn-in testing (in the order of weeks) in order to ensure the system had strong reliability – an absolute necessity given the deployment domain. Immediately following testing, the prototype systems were deployed at two locations. Once deployed, feedback from users was quickly received and used to drive minor modifications, e.g. blocking of senders due to offensive messages.

2.4.3 Summary of Development Strategy

Initial requirements gathered at a design workshop, using rapid prototyping of scenarios (rather than hardware/software) with potential users. Rapid prototyping using mainly 'off the shelf' hardware and software components in order to produce a system ready for deployment quickly. Simple user interface development based on requirements from the design workshop. Logging facilities included in the prototype for it to act as a technology probe. Requirements for high system reliability meant careful consideration of technology used and considerable time spent on 'burn-in' testing.

2.5 Intelligent Office User Interface

The Intelligent Office system [10] controls various electrical appliances (such as a fan, heater and lamp) by sensing environmental context and by making proactive suggestions. These suggestions are based on rules automatically 'learnt' from a context history or input by the user e.g. if the temperature is 'hot' and the fan is off, prompt to turn on the fan. However, the adoption and use of such a system raises many HCI challenges which had previously not been addressed.

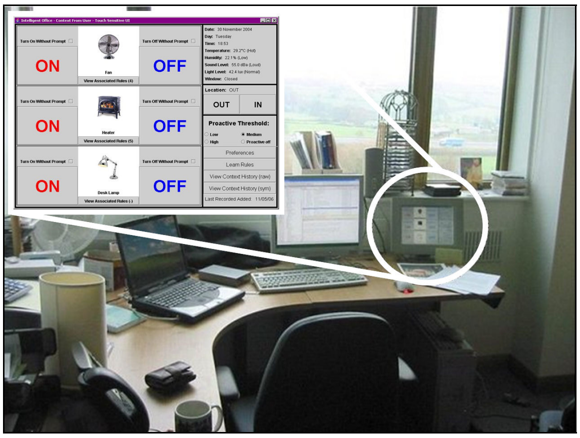

Figure 3 - Intelligent Office Deployment and User Interface (insert)

2.5.1 Aims

The primary goal of this work was to investigate how the reasoning behind the proactive behaviour of an intelligent office system could be made more visible to users, helping them to understand the system's behaviour. This required a major redesign of the user interface in order to provide a peripheral interactive desktop display (figure 3), enabling manual control of appliances, presentation of information and a platform for exploration of the system's original goal.

2.5.2 Approach

The initial approach was that of rapid prototyping and small scale deployment to gather requirements. This process was guided by feedback from the previous user interface. Development primarily involved only the user interface, leaving the intelligent office system unchanged. A questionnaire was used to help generate requirements for this system, presenting questions in the context of possible scenarios.

2.5.3 Summary of Development Strategy

Rapid prototyping of a user interface for an intelligent office control system, in order to investigate the visibility of decisions made by the system to the user. High-fidelity prototyping to demonstrate the concept to potential users, in order to gather feedback and uncover requirements. Rapid prototyping a user interface on top of a working underlying system. Use of a questionnaire in order to generate requirements.

3. Issues

Building a novel 'ubicomp' system typically involves dealing with issues on a wide range of levels, often due to tailored hardware and specialised software solutions. In this section we discuss key issues which we have encountered in our use of rapid prototyping.

3.1 Tailoring 'Off the Shelf' Technology

Many times during our work we have sought to tailor ordinary 'off the shelf' technology in order to build our novel 'ubicomp' prototypes (some examples include GSM modems, PDA devices, PC hardware and a games console). On the surface this appears a practical solution, making use of readily available technology often with proven reliability at modest development cost. However, we have found that using this type of technology in ways which may not have been intended by the original designers can have severe implications. In several cases we have found areas such as operating system and programming language support missing or incomplete. Tailoring often requires extensive investigation and testing on the part of the developer, greatly reducing the rapidity with which prototypes can be produced.

When developing the Hermes Photo Display, enabling Bluetooth interaction using mobile phones proved far more time consuming than expected. After spending several weeks considering and testing various options, the only solutions providing the appropriate level of support and flexibility required was to use specific hardware, operating system and programming API.

When attempting to support authentication using iButton readers on the Hermes 1 door displays, we discovered a complete lack of suitable solutions to support the required Java Communications API on the platform we used. After spending weeks attempting find an appropriate solution, a commercial solution became available. Unfortunately, after deploying prototype door displays with iButton support, we found this adversely affected the reliability of the door display clients, leading to this idea effectively being abandoned.

While tailoring of existing 'off the shelf' technology can seem appealing, and in some cases prove a very useful approach, we see its use as a trade-off between the pitfalls of attempting to tailor existing technology and investing time and effort in a specialised solution. This trade-off must be considered carefully, especially in circumstances where domain knowledge may be limited or unavailable.

3.2 Reliability in the Context of Rapid Prototyping

The systems described in section 2 have high reliability requirements (but not necessarily at all phases of development, e.g. Hermes 1 phase one) and this contrasts with the usual rapid approach of short term demonstration or testing. We have had to deal with the issue of reliability many times and it has raised challenges on a range of levels.

In the Hermes 1 system where door displays were running 24/7 continually, we found that display devices had unreliable operating system drivers and software components, occasionally causing a door display to crash after several weeks. This has implications for:

- Housing – In the early stages of the project we were forced to forgo some security features of the housing in order to enable us remove and reset the door display device (a PDA) easily.
- Hardware Platform – In order to increase security, door display devices had to be modified to route the reset switch outside of the case.
- Operating System and Software Platform – Door displays had to be temporarily decommissioned in order to apply updates and change configuration settings to improve reliability.
- Application – Sending of a regular 'heartbeat' from the client to the server was added in order to determine whether a device was 'alive' or not. This was then used to inform a system administrator and the affected user via e-mail.
- User Interface – Occasionally a door display would crash, but still display a user interface making it appear as if it was still working, confusing users and reducing trust.
- Human Factors – If someone wishing to leave a message on a door display doubts whether their message will actually reach its owner, he or she will simply use a post-it note instead.

During the burn-in testing of the SPAM system we discovered that the GSM modems we used appeared to have a fault, occasionally causing them to stop responding until reset. Unable to replace these components at this late stage, the display systems were configured so that a single press on the power button would shut down the display application and power off, then when powered on would automatically start the application and restore the previous state. This feature was understood by users of the SPAM system, being analogous to other information appliances such as a mobile phone, which occasionally crash and need to be turned off and back on in a similar manner.

The Hermes picture display is based around a Philips DesXcape Smart Display (a wireless Microsoft Windows XP Remote Desktop client) which enabled us to rapidly prototype and deploy a display system, requiring only power and wireless network coverage. However, on several occasions, we found disruption to the wireless link caused the display to unexpectedly disconnect from the Remote Desktop host, also the additional overhead caused by the Remote Desktop connection seemed to make the host machine less stable. The use of a Smart Display in this case demonstrated a trade-off between the need for reliability and the need for easy and fast deployment: the requirement for user feedback from experience of a high-fidelity prototype outweighed the minor reliability problems.

We have come to the conclusion that rather than attempting to provide very high levels of reliability, a more practical approach is to provide notification of failure (as mentioned previously) and to manage user expectations.

3.3 Managing User Expectations

While encouraging users to "suspend their disbelief" (meaning the removal of any doubt about the reality of a situation) is an accepted part of low-fidelity prototyping, the same cannot be said of our prototype systems deployed outside of the 'lab' (i.e. for everyday use). In these circumstances the majority of users' expectations of our prototypes were that of a 'finished' product. Perhaps this is to be expected, when we provide a system that is integrated into existing working patterns, effectively implying high reliability even though the system is only an experimental prototype.

A crucial aspect of deploying in the 'real world' is managing these user expectations. If one considers the development and maintenance of trust as a staged process [11] [12], we have tried to ensure that the range of deployment only grows in-line with system reliability, rather than risk damaging the important initial trust-establishment phase. On occasion this has been 'out of step', for example during the later deployment phases of Hermes 1 reliability problems did affect user trust and for some users we found the task of re-establishing trust challenging – despite strong improvements in the overall reliability of the system.

3.4 Fostering a User Centred Approach

During our work we have found it useful to use more traditional ethnographical methodologies to 'jumpstart' our user-centred rapid prototyping process, gathering user requirements before the development process begins (as we are doing with Hermes 2) to improve the usability and acceptance of initial prototypes. This effectively means that fewer rapid prototyping cycles are needed at the beginning of the development process. The use of a participatory design workshop in the SPAM system enabled the development of a successful initial prototype, which very nearly met all requirements for its users.

One issue that has arisen time and again is the need as a designer to be sympathetic to user's patterns of use. For example, during the early phases of development in Hermes 1, we forced owners to interact with their door display through a web browser. This caused a problem for two reasons. Firstly, it was often inconvenient for users to open a web page and enter their user name and PIN in order to set or read a message. Secondly, and perhaps more importantly, the approach did not fit in with the way many users seem to process the task of leaving a message. It was only because we involved users in the design of the system that we could develop approaches that fitted with their existing patterns of work, for example providing MSN messenger and e-mail integration.

3.5 Deployment for Understanding Domains

The notion of Technology Probes has recently been employed in the Interliving Project [13]: adaptations of

Cultural Probes that seek to embed inspiration within the design process. These Technology Probes situate existing technologies in real homes rather than 'lab houses' in order to inspire design by exposing inhabitants to new experiences. While our use of prototype showcases does serve to inspire feedback from users, we also use the notion of a Technology Probe in a different and perhaps simpler way: the embedding of a logging system into the technology itself. Both Hermes 1 and SPAM included logging of user interactions to act as a 'probe'. Analysis of these logs improved understanding of these two domains to help drive the prototyping processes and allowed various aspects of use to be quantified.

4. Lessons Learned

Table 1 summarises the different prototyping techniques we have used in our user-centred rapid prototyping approach, along with the systems for which they were used. In addition, the table lists the key advantages and disadvantages that each individual technique entails. It also illustrates that we applied several techniques within each individual project, which has led to two broader central lessons we now discuss.

4.1 Multi-phased Prototyping

In the majority of cases the faster or earlier a prototype is deployed the faster feedback and new requirements can be obtained, and the design improved. However, early prototypes are inherently unfinished and may have errors, which, if expectations are not managed, may negatively impact user attitudes towards the system and make user-centred design difficult. There is clearly a trade-off between the need for feedback and the need to maintain use or encourage adoption when prototyping 'ubicomp' systems.

As a consequence, we have adopted a *multi-phased prototyping approach,* driven by user-centric considerations. We apply different prototyping techniques at different stages of a project to gather user feedback about system properties, contextual factors, and concerns they may have. We then iteratively validate and refine a prototype until it meets the users' expectations and occasionally introduce new features to explore further options. In the case of prototype deployment, we distinguish phases in terms of the user groups, exposed to the prototype. We incrementally move from fault-resilient users (such as system developers) to 'sympathetic' users (such as people close to the development) before deploying it to 'real-world' users who may not tolerate any misbehaviour of the prototype. In combination with a longitudinal perspective (i.e. evaluating prototypes for several months instead of just a few days) this approach has enabled us to gather more information than would have been possible otherwise.

Examples that might have gone unnoticed using a conventional prototyping approach include the impact of the iButton hardware unreliability on user acceptance, usage issues with the initial user interface in the Intelligent Office project, or the requirement for a blocking feature in SPAM (see section 2 and table 1).

4.2 Technical Feasibility and Use

While rapid prototyping is an excellent approach for investigating the technical feasibility of 'ubicomp' systems, this investigation can be meaningless without studying real use. In our work, exposing users to prototypes in order to generate feedback has been crucial to enable use. For example, the original Intelligent Office prototype demonstrated that such a system was technically possible (sensing environmental context and making proactive suggestions). However, after exposing the system to users we found it was simply too difficult to use on a regular basis, the interaction model simply did not fit the desktop environment in which it was used (prompting the development recounted in section 2.5).

Another trade-off is in the use of 'off the shelf' technology, which can provide benefits such as reduced development time and 'jumpstarting' reliability (i.e. using reliable building blocks). However, this type of technology typically has to be tailored to a specific purpose, and we have found that use beyond that intended by its designers can be a difficult and challenging exercise. When developing the Hermes Photo Display we found that the Bluetooth discovery process is relatively long and often unreliable; during a user study with over ten devices in range, the server had trouble discovering more than five or six at once.

Technique	System	Key Advantages	Key Disadvantages
Prototype showcases (demonstrating 'collections' of potential prototypes to end-users)	**Hermes 2**	• supports early user choice • generates useful feedback – users can signal the 'best bits' of the different prototypes presented	• time consuming to organise, design and build potential prototypes • difficult to pre-empt all the concerns users may have
Participatory design workshops (supporting the user-centred design of both early and more mature prototype systems)	**SPAM**, (*design workshops associated with **Hermes 2** and specific deployments of the **Hermes Photo Display** to take place shortly*)	• helps users feel truly 'involved' in the user design process • props/prototypes can be used to inspire discussion	• cost associated with organisation of the workshop
Paper prototyping (demonstrating early user interface designs on paper to end-users)	**Hermes 2**	• low cost • high flexibility • helps include users early in the design process	• can be difficult to provide sufficient fidelity for a user to appreciate the scenario.
Questionnaire-based user Studies (obtaining feedback on early/more mature prototypes)	**All**	• low cost of setting up • high flexibility	• time consuming for user to complete a questionnaire properly, can be difficult to investigate the rational for a participant's response
Lab-based testing of early prototypes	**All**	• efficient and useful for ascertaining initial technical feasibility • useful for 'burn-in' reliability testing	• open to requirements capture problem
Early deployment of Prototypes – **'Real world' users** (unbiased)	**SPAM**, **Hermes 1** (second phase onwards, where deployments involved end-users not directly associated with the project), **Hermes Photo Display** (second phase of deployment)	• users can experience 'real world' use of the prototype regular basis which generates very useful feedback	• difficult to maintain reliability, therefore trust, and therefore regular use • usually requires some form of management and user support
Early deployment of Prototypes – **'Friendly' users** (connected with the work and therefore sympathetic)	**Hermes 1** (initial phase where units were only deployed outside the offices of developers), **Hermes Photo Display** (initial phase of deployment), **Intelligent Office,** *(will also be used in **Hermes 2**)*	• users can experience real world use with a potentially unreliable prototype without a large impact on trust (and use) • generates useful feedback quickly	• sometimes difficult to involve a range of users, therefore potentially open to the requirements capture problem • sympathy is likely to affect judgement, so feedback may be biased/inaccurate
Cultural Probes (obtaining ethnographic data on usage)	**SPAM**, **Hermes 2,** *(will also be used for specific deployments of the **Hermes Photo Display**, e.g. to support specific societies/communities at Lancaster University)*	• helpful insights into use • relatively easy and fast to set up • low financial cost	• may not capture all use due to practicality of logging for participants
Deployed prototypes as technology probes (integrating logging facilities to help understand actual use and therefore help drive further developments)	**Hermes 1**, **SPAM,** *(will also be used for **Hermes 2** and certain deployments of the **Hermes Photo Display**)*	• easy to collect data • relatively easy to analyse and quantify use	• implications for design, analysis may include the time-consuming manual tagging of data

Table 1 – Summary of Techniques Used

4.3 Concluding Remarks

The use of prototyping has been critical to our user centred and multidisciplinary exploration of interactive display-based systems. In addition to using traditional approaches such as questionnaires, semi-structured interviews etc. we also received valuable information by incorporating logging functionality into the prototypes themselves (as well as the final deployed system). In this respect, the deployed prototypes have acted as technology probes. Typically the requirements of a given prototype vary throughout the development process and prototyping strategies should reflect this.

In this article we have reported on our experiences developing several interactive display-based systems and have presented a detailed analysis of a number of prototyping techniques used. We hope this will help other researchers and practitioners to select an appropriate prototyping technique for future projects. At the heart of these techniques has been a strategy based on multi-phased prototyping, which calls for incrementally applying different techniques during the whole project development life cycle and the incremental deployment of prototypes in different user groups.

Acknowledgements

This work is funded by the EPSRC funded CASIDE project (grant ref: EP/C005589). The work also builds on work carried out under the EPSRC funded Equator and CASCO projects. Hardware support for future trials based on the work described will be provided by the SRIF II (Science Research Investment Fund) funded e-campus initiative.

References

[1] Weiser, M., "The computer for the 21st Century", Scientific American, Col. 265 No. 3, pp. 66-75. 1991.

[2] McCarthy, J. F., Costa, T. J., and Liongosari, E. S. UniCast, OutCast & GroupCast: Three Steps Toward Ubiquitous, Peripheral Displays, In proceedings of Ubicomp 2001, (Atlanta GA, September 2001), 332-345, LNCS 2201, 2001.

[3] Markus, M. L., and Connolly, T., Why cscw applications fail: Problems in the adoption of interdependent work tools. In proceedings of CSCW'90, pp. 371 – 380, ACM Press, 1990.

[4] Cheverst, K., D. Fitton, and A. Dix. "Exploring the Evolution of Office Door Displays", in: Public and Situated Displays: Social and Interactional aspects of shared display technologies. K. O'Hara, M. Perry, et al (Eds). Chapter 6, pp. 141-169, Kluwer. ISBN 1-4020-1677-8. 2003.

[5] Cheverst, K., K. Clarke, D. Fitton, M. Rouncefield, A. Crabtree and T. Hemmings "SPAM on the Menu: the practical use of remote messaging in community care.", In proceedings of CUU'2003, Vancouver, Canada, ACM Press, ISBN: 1-58113-701-X, pp. 23-29, Nov. 10-11, 2003.

[6] Gaver, W., Dunne, A., Pacenti, E. Design: Cultural Probes, In Interactions: New Visions of Human-Computer Interaction. 6(1), pp. 21-29. 1999.

[7] Mankoff, J., and Schilit, B., Supporting knowledge workers beyond the desktop with PALPlates. In proceedings of CHI '97, pages 550--551. ACM Press, 1997.

[8] Cheverst, K., Dix, A., Fitton, D., Kray, C., Rouncefield, M., Sas., C Saslis-Lagoudakis, G., Sheridan, J., Exploring Bluetooth based Mobile Phone Interaction with the Hermes Photo Display. In proceedings of Human Computer Interaction with Mobile Devices and Services (Mobile HCI), to appear, 2005.

[9] Greenbaum, J., Kyng, M., (eds) "Design at Work: cooperative Design of Computer Systems" Lawrence Erlbaum Associates, Hillsdale, N.J., 1991.

[10] Cheverst, K., H.E. Byun, D. Fitton, C. Sas, C. Kray and N. Villar, "Exploring Issues of User Model Transparency and Proactive Behaviour in an Office Environment Control System." Special Issue of UMUAI (User Modelling and User-Adapted Interaction) on User Modeling in Ubiquitous Computing, Kluwer, to appear, 2005.

[11] Siau, K., and Shen, Z. Building customer trust in mobile commerce. Communications of the ACM, 46, 4, 91-94. April, 2003.

[12] Sillence, E., Briggs, P., Fishwick, L., and Harris, P. Trust and Mistrust of Online Health Sites. In proceedings of SIGCHI Conference on Human Factors in Computing Systems (CHI '04) (Vienna, Austria, April 24-29, 2004). ACM Press, New York, USA, 663-670, 2004.

[13] Hutchinson, H., Mackay, W., Westerlund, B., et al. Technology Probes: Inspiring Design for and with Families. In proceedings of CHI'03. ACM Press, pp. 17-24, April, 2003.