

# HOW TO DESIGN AND PROTOTYPE AN INFORMATION APPLIANCE IN 24 HOURS – INTEGRATING PRODUCT & INTERFACE DESIGN PROCESSES

Steve Gill Gareth Loudon Bethan Hewett Gareth Barham

National Centre for Product Design Research, University of Wales Institute, Cardiff Email: [sgill@uwic.ac.uk](mailto:sgill@uwic.ac.uk)

**ABSTRACT:** *This paper examines a number of current methods for prototyping Information Appliances. It then reports on the results of a 24 hour product design exercise held at the National Centre for Product Design Research (PDR) based at the University of Wales Institute, Cardiff, (UWIC) in partnership with Nottingham Trent University, Audi Design Foundation and two UK Top Ten design consultancies.*

*It concentrates on the activities of the interface design team based at UWIC, their design strategies, and, in particular, the use of their design & prototyping system. It also reports on the structure of the 24 hours, negotiations between the product and interface design teams, the consequent concessions and the integration of Graphic User Interface (GUI) and hardware aspects of the interface design process.*

*In conclusion, the interface development methodology, its employment, and its potential commercial use are evaluated.*

**KEYWORDS:** Interaction, Design, Information Appliance, Product, Prototyping

## 1. INTRODUCTION

There are a number of methods currently used by designers to design, develop and test Information Appliances. This paper will critically evaluate these, and, by means of a case study, examine ways in which the best of these methods might be combined with new techniques being developed at the National Centre for Product Design Research (PDR) in UWIC.

## 2. CURRENT METHODOLOGIES

A few examples of best practice in Information Appliance development are presented below in

three discrete categories, *Low Fidelity*, *Medium Fidelity* and *High Fidelity*. Each method has its place, its strengths and its weaknesses. This paper will briefly examine, describe, and evaluate these. (For case studies from *IDEO* see Spreenberget al 2002).

### 2.1 Low Fidelity Prototyping

The design process has a number of stages, each of which builds upon the last, each decision having a direct effect on all that follow. The first stage of a good design process involves the fast and “fluid” development of a number of ideas which, by their nature, are underdeveloped. These ideas should be swiftly evaluated and examined for large, fundamental flaws. It follows that any process used at this stage must be capable of being deployed very quickly, enabling a high volume of ideas to be analysed. Low fidelity prototyping is therefore the most appropriate method to use during this stage. Three methods of producing rapid, low fidelity prototypes are briefly discussed below.

#### 2.1.1 State Transition Diagrams

One of the quickest methods of exploring the design of an Information Appliance is to sketch its various states on *Post-It* notes (Gill 2003). The control that activates a state change is circled and the new state is drawn on another *Post-It* with an arrow linking the two (see *Figure 4*). Using *Post-Its* allows the designer to evolve the design by moving or removing elements as better ideas are formulated. *Post-It* state transitions are quick, cheap, and can be used in brainstorming sessions within the team or in broader contexts. They do not, however give any sense of the appropriateness of the product’s form

and a lot of imagination is required to visualise the user's experience.

### 2.1.2 "Soft" mock-ups

"Soft" prototypes (see *Figure 1*) are often combined with paper prototypes containing the *GUI* ideas and a three dimensional representation of the Information Appliance (Sprenberg et al 1995).



**Figure 1** "Soft" model with replaceable Post-It note screens

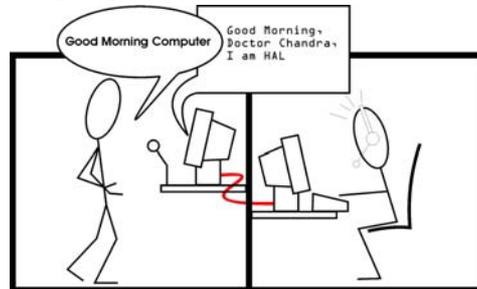
This method sacrifices some of the speed and "fluidity" of the purely paper-based system but enables users to get a feel for the form and ergonomics of the Information Appliance including the placement and appropriateness of its control inputs. Users "trigger" a control input and the designer accordingly chooses an appropriate *Post-It* note displaying the correct State of the Information Appliance to place on the "screen". This works for products with simple physical control inputs but does not adequately allow simulation of more sophisticated input methods like speech to text or handwriting recognition.

### 2.1.3 Wizard of Oz experiments

One common way of simulating more sophisticated input types is the *Wizard of Oz* method (Maulsby et al 1993). This technique involves placing the user in front of a computer in one room and an operator in another (see *Figure 2*). The operator is able to see and/or hear what the user is doing to effect inputs, and simulates the machine's response by inputting the required data by an appropriate method. To the user, it appears as if the machine is responding directly to their inputs.

While all of these techniques are useful to roughly

"debug" an idea, their common problem is the difficulty of capturing the product and user's true behaviour. Recreating the subtleties of behaviour and context (which can be of vital importance in designing a product interface) are additional issues.



**Figure 2** Wizard of Oz experiment

## 2.2 Medium fidelity prototyping

### 2.2.1 Storyboards

One method of helping the designer to understand the context of a product's environment is the storyboard. Storyboards have two major manifestations, paper-based and multimedia. In either case the storyboard highlights the designer's ideas for the product itself and its interactive elements by telling a story based around the product's intended context. The story relies on the product's features and functions as triggers for its plot. A storyboard's value is both in forcing the designer to consider the product in its context and its ability to communicate effectively to a third party, enabling a useful dialogue between the designer and intended user.

The problem with storyboards is that while they contextualise an issue and force the designer to consider broader issues, their very nature means that the all important *unexpected* occurrences when a user interacts with a product lie undiscovered.

### 2.2.2 Façade prototypes

Façade prototyping is a method whereby a product is simulated in two dimensions on a computer screen. Typical tools for creating this type of prototype include *HyperCard*, *Director*, *PowerPoint*,

*Flash* and *Denim* (Newman et al 2003). The advantage of façade prototypes is that they allow the user to interact directly with a representation of the product. Their disadvantage is that the representation is two dimensional, meaning that in many cases much of the information gathered will be distorted because it is often difficult to contextualise use, and the user is interacting with a mouse or touch screen rather than the actual product. Sharp (1998) demonstrated that virtual prototyping of touch-screen GUIs could be effective this way, but the product in Sharp's study was a microwave oven with a flat vertical surface. The user interacted with it in the way they would a touch-screen, so a 3D product was not, in effect tested.

## 2.3 High fidelity prototyping

### 2.3.1 Experience Prototyping

*Experience Prototyping* aims to simulate real situations experienced by users interacting with a product, space or service as closely as possible. Subtle design issues are teased out by creating scenarios using props, models and actors then monitoring the user and their actual or potential interaction with the product (Buchenau and Suri 2000). This technique is very good for gathering detailed information that informs the design process in ways that other tests do not and can be used in conjunction with any of the methods mentioned above.

### 2.3.2 Hardware/Software prototyping

Several companies use prototyping techniques that combine software and hardware prototyping (*IDEO*, *Philips*, *Motorola*, and *Handspring* being examples). An example of this type of technique is that used by *Sony Ericsson*.

The *Sony Ericsson T610* has a mode that allows it to be linked to a PC for certain procedures. *Sony Ericsson* created a programme that converted a *T610*'s outputs from each of its controls to

*American Standard Code for Information Interchange (ASCII)* character codes. They had, in effect, created what *Handspring* refer to as the 'Buck System' (Pering 2002). The difference in this case is that re-mapping of the controls is not a problem as Pering describes with the 'Buck'. Because the hardware delivers *ASCII* codes, the designer simply alters what the code triggers in the software to effect a change.

Although this is a useful system that works very well in some contexts, it is not as flexible as designing the entire software/hardware package together. This is not a particularly simple issue. As Pering noted, one reason hardware and software were not developed in tandem at *Handspring* was that hardware development was so much slower and more expensive than software development.



**Figure 3** *Sony Ericsson T610/PC* combination

What is required is a method by which both software and hardware may be developed rapidly, in tandem, and in a manner that is "designer friendly". This is what the team at *UWIC* have been working to produce.

## 3. DESIGNING AN INFORMATION APPLIANCE IN 24 HOURS

The concept and management of the project to design and prototype a product in 24 hours are described in more depth in Griffiths (2004). For a comparison of some of the prototyping and development techniques used with more traditional methods see Gill *et al* (2005).

The aim of the project was to design and prototype an Information Appliance in 24 hours. The interface

design team felt that this was an ideal chance to use the *IE method* developed at *UWIC*, a system that produces prototypes of Information Appliances by linking models to a PC like the *Handspring* “Buck”, but that allows very rapid development and a high level of flexibility in a project’s early stages. The system can be used for medium or high fidelity outputs. The team elected to produce the latter.

### 3.1 Project management

The project team was divided into four groups covering product design, interface design, CAD work and prototyping/modelmaking. Interface design and prototyping matters are given precedence in this paper. Some of the techniques used and their background are described in greater detail in Gill (2003).

**3.1.1 8.00 a.m. – Briefing: Design a “blue sky” communications device for 18 - 25 year olds interested in extreme outdoor pursuits.**

**3.1.2 8.10 a.m. - *Creative Problem Solving technique (CPS)*, (Van Gundy, 1988) and *Six Hats idea evaluation* (De Bono (1990)) run simultaneously**

The main conclusions were that a “safety product” would not be suitable and that competitive and social elements should be included.

**3.1.3 9.30 a.m. – Review of creative processes and research data arrives at a product concept**

The two concept generation teams were merged to discuss their conclusions, and more detailed discussions took place informed by reports from the research team regarding new materials plus the information gleaned from the target user group. A large part of the discussion revolved around the functions of the product and the wishes and habits of the user. Control inputs were discussed at this stage but the overall look of the product was not.



**Figure 4.** *Post-It* State Transition Diagram

**3.1.4 10.40 a.m. – Concept selected**

**3.1.5 11.00 a.m. – CAD development commences. User interface team begin designing**

A user interface team of four set about conceiving an interface that would deliver to the user the functionality conceived by the design team as a whole. It was agreed that the major control input device should be a rotary dial because it would be easy to see the current function state by looking at the dial’s position. This decision had important implications for the product’s design. A representative from the interface design team negotiated changes in the design with the product design team. Further communications were required as the number of functions, their sequence, and the designs of symbols were decided.

**3.1.6 3.00 p.m. – Work commences on prototyping the GUI**

When the form and basic functions of the product were decided, the interface design team began detailed design work. The first step in this process was the production of a *Post-It* state transition chart which effectively became the design specification of the interface prototype. This technique was used exactly as described in 2.1.1 except that in the *UWIC* method each note is given a number (for reasons that will become apparent below in 3.1.8).

**3.1.7 4.00 p.m. – External surface CAD data sent to interface prototype team for internal design to commence**

With the external surfaces of the product fully

detailed the interactive prototype designers and makers could start work. Their job was to create a model which would link up to the GUI prototype allowing potential users to gain a good feel for how the completed product would be to actually use.

### **3.1.8 6.00 p.m. – GUI States prototyped, work commences on linking them and on detailing the GUI**

When the State Transition Diagram was completed, the state on each numbered *Post-It* was reproduced as a *PowerPoint* slide. The direct correlation between the numbered *Post-Its* and the *PowerPoint* “slide” it represents means that the designer is able to use the State Transition Diagram to lay out the detailed *GUI* design. Each of these states frequently requires further work in the *PowerPoint* prototype that would not have been possible to represent on



**Figure 5.** Interactive model driving PC-based GUI the State Transition Diagram (for instance speech to text, text to speech, animations, or sound).

### **3.1.9 12.00 a.m. – GUI is completed and keyboard enabled**

So far *PowerPoint* had been used to create a façade prototype (2.2.2). To progress it, *PowerPoint*'s embedded language, *Visual Basic for Applications (VBA)*, was used to make hyperlinks linking each of the states of the interface together. The control input triggering each of these changes in state is effected by a QWERTY keyboard press. The end result is a PC-based simulation of the GUI controlled via the keyboard.

### **3.1.10 1.00 a.m. – Manufacture of interactive model commences**

When the interactive prototype CAD work was complete, the data was used by a CNC machine to manufacture a prototype from solid blocks of model board.

### **3.1.11 6.00 a.m. – CNC components hand-finished. Interactive prototype wired up.**

In order for the interactive prototype to connect to the PC-based GUI prototype, switches had to be embedded in it. These were wired and tested before being mounted within the model. When connected to a product called an *IE Unit* (Gill 2003) they allowed the model to simulate keyboard inputs (see *Figure 5*). Thus when a user activates a switch in the model, the PC behaves as if it has received a keyboard input and so the GUI prototype responds accordingly.

### **3.1.12 7.30 a.m. – Interactive prototype completed. Testing commences.**

### **3.1.13 8.00 a.m. – Finish**

The team's answer to the brief was *Mohawk*, a device for extreme sports enthusiasts of the 'PlayStation Generation'. The device allows users to record their performance in a “real world” activity (in this case mountain biking), publish it to the internet, keep track of their “tribal” and world rankings, and challenge and meet others from their “tribe”. Users could even race “virtually” because the device would have the ability to “ghost” an image of a competitor's experience on top of their own.

## **4. CONCLUSIONS**

As discussed above, there are already a number of methods for developing Information Appliances, each with their strengths and weaknesses. The system described in the case study above should be regarded as another tool to be used in conjunction

with current methods as appropriate.

The 24 hour product design exercise proved that it was possible to conceive, detail and prototype an Information Appliance within a very short period of time using this system. Comparisons with the techniques used by a second team (Gill *et al* 2005) suggested that more could have been gained from the process had it been used as a medium fidelity as opposed to high fidelity prototyping tool. The extra time could have been spent to conducting user testing, possibly using Experience Prototyping techniques. The tool is quite capable of being used in this fashion and given the time restraints, this should clearly have been the method adopted by the team.

#### 4.1 Potential for industry

The potential for the system's use in industry is clear. Unpublished research commissioned by *UWIC* has revealed a great deal of interest from both the product and interface design community. It is apparent from this research that the system's greatest appeal will be in the medium and high fidelity prototyping areas. To this end the team has been actively collaborating with *Sony Ericsson*, *Samsung Design Europe* and *Alloy Total Product Design* to produce a viable solution, and may invite other collaborators to join soon. It has become clear, both from these collaborations and the 24 hour product design exercise, that the key to the system's success will be its ability to increase design flexibility and depth while reducing prototyping times. In terms of medium fidelity prototyping, *UWIC*'s aim is to reduce the time taken to prototype from a "blank sheet" to one hour or less.

#### REFERENCES

- [1] Buchenau, M. and Suri, J.F. (2000) *Experience Prototyping*, DIS '00, Brooklyn, NY
- [2] De Bono, Edward, *Six Thinking Hats*, Penguin, England, 1990
- [3] Gill, S. (2003) "Developing Information Appliance Design Tools for Designers" The Proceedings of the 1<sup>st</sup> Appliance Design Conference, May 2003, HP Labs, Bristol, UK
- [4] Gill, S., Johnson, P., Dale, J., Loudon, G., Hewett, B. and Barham, G. (2005) "The Traditional Design Process Versus a New Design Methodology: a Comparative Case Study of a Rapidly Designed Information Appliance" The Proceedings of the Human Oriented Informatics & Telematics Conference 13<sup>th</sup> – 15<sup>th</sup> April 2005, University Of York, York, UK
- [5] Griffiths, R., The 24-hour Product – From Concept to Interactive Model In Less Than a Day, Proceedings of The 2nd International Engineering and Product Design Education Conference September 2 – 3, 2004 Delft, The Netherlands
- [6] Maulsby, D., Greenberg, S. and Mander, R. (1993). Prototyping an intelligent agent through Wizard of Oz, Proceedings of the SIGCHI conference on Human factors in computing systems, Amsterdam, The Netherlands
- [7] Newman, M.W., Lin, J., Hong, J.I., and Landay, J.A. (2003) "DENIM: An Informal Web Site Design Tool Inspired by Observations of Practice." *Human-Computer Interaction Journal* 2003. 18(3): pp. 259-324.
- [8] Pering, C. (2002) Interaction Design Prototyping of Communicator Devices: Towards meeting the hardware – software challenge" *Interactions Journal* November and December 2002
- [9] Sharp, J. (1998) *Interaction Design for electronic products using virtual simulations*, PhD Thesis, Brunel University, UK
- [10] Spreenberg, P., Saloman, G. and Phillip, J. (1995) "Interaction design at IDEO Product Development", The Proceedings of the Conference on Human Factors in Computing, May, 1995, Denver, Colorado
- [11] Spreenberg, P., Saloman, G. and Phillip, J. *Information Appliances*, chapter from *The Human Computer Interaction Handbook*, Julie A. Jacko, Andrew Sears (eds), 2002
- [12] Van Gundy, A. B. (1988). *Techniques of structured problem solving*. New York: Van Nostrand Reinhold