ARTIST-DESIGNERS AND ENGINEER-DESIGNERS

G Crampton Smith

Rapporteur: Steve McGough

ARTIST-DESIGNERS AND ENGINEER-DESIGNERS

Gillian Crampton Smith
Computer Related Design
Royal College of Art
Kensington Gore, London, SW7 2EU
g.crampton-smith@rca.ac.uk

Readers may be surprised to find a paper from an art school in a collection of papers about software architecture and design. What, you may ask, have artists got to do with software? In this paper I will talk about the contribution of artist-designers—as opposed to engineer-designers—to the design of interactive systems and products. I will discuss why interaction design is becoming important now and describe its four elements: the design of function, representation, values, and user-experience.

In fact art schools are not only populated by artists, they are also the home of designers—graphic and industrial designers, architects or fashion designers, to name but a few. Their focus is as much on things that fulfill people's desires as on things that fulfill their needs. Their art is to make things that that *work* for people as well as *delight* them, making things that move people aesthetically and emotionally—thrill them, satisfy them, calm them, reassure them, delight them—shaping emotion in ways appropriate to the users and their context. This is not something particular to late twentieth-century developed nations: objects and adornment have symbolic and ritual significance in every society; the culture of artefacts is is as important to tribespeople far from technological development as it is to people in highly developed consumer societies like our own.

The Royal College is Britain's graduate school of art and design with 800 students in 25 different disciplines from the fine and applied arts, design, communication, and theory; and though it is called the Royal College of Art, 75% of its students are in design disciplines. Set up in the nineteenth century, its charter was to bring 'art to manufacture'. A hundred years ago this was the industrial manufacture of physical things and then, as now, the British were worried about their competitors making more useful and more desirable products than they were. Today the College also designs immaterial things, the form and experience of interactive systems and products—interaction design.

Interaction design

The Computer Related Design Programme at the RCA grew out of a CAD programme in the Industrial Design department. However, as soon as designers sat in front of these rather complex systems, they immediately started to think: why do they have to *be* like this? couldn't they be designed differently? And so people began to explore ways traditional design knowledge and skills, from industrial or graphic design, architecture, fashion, or film, could be applied to designing interactive systems.

My own background was in graphic design and in the early 1980s I designed and written a program to do page layout on the screen—early desktop publishing. As I worked on it, I could see many parallels with the design of information in traditional media. I expected, mistakenly as it turned out, that graphic designers would soon become involved in software design—not at that time because they could make it pretty (because with character-based displays that was hardly an option) but because they could make it comprehensible and transparent to users in the way that a well-designed book is transparent, enabling readers to see through the form to the ideas it contains. I hoped that good software design could similarly allow people to 'see through' the software to concentrate on *what* they are doing, rather than *how* they are doing it.

The Computer Related Design programme became my responsibility in 1990 and I decided to take people from all kinds of art and design backgrounds who were prepared to explore ways their discipline might relate to the design of interactive systems. (We also took one or two engineers and psychologists who had already worked in the computer industry.) It was clear that in principle artist-designers had knowledge and skills to contribute, but not at all clear how to do it in practice. Discovering this became the aim of the programme.

In the beginning the focus was on software design, but as many students had 3-D design backgrounds, it naturally moved to products that spanned the physical and the virtual, like hand-held computers or intelligent furniture.

Links with industry were important from the start: Apple computer was an early sponsor, and the department exchanged staff with the Advanced Technology Group. Four years ago it began to collaborate with Interval Research in Palo Alto, which allowed it to start a research group and work on projects more long term. CRD now has a range of industrial sponsors who are interested in its approach to the design of computer-based products and systems and is a partner in two European collaborative projects projects on intelligent information interfaces, one to do with mobile phones and the other exploring the potential of ITT to connect the elderly better into society.

So the department was founded on two premisses: firstly that computing systems could be different, and a lot better from the user's point of view. And secondly that as computing systems are beginning to permeate the whole of life, the same requirements of civility, grace, decorum that we require from other parts of our life should apply to these too.

Why now?

The contribution of artist-designers is becoming more important now for two reasons, the first to do with the development of technology and the second with its changing market.

The past fifteen years has seen changes in technology that both enable and require much more careful design of products for the people who will use them:

Colour displays now allow much richer visual communication than before. More complex information can be designed, and in more expressive ways, but it is more difficult to do. Consider the green screens of fifteen yeas ago, where the only variables you had to play with were position on the screen, black type, reversed type and flashing type—and if you were lucky, lowercase letters too. Today's graphic screens in comparison provide the depth and complexity of a printed book, combined with the immediacy and production values of a television programme.

Miniaturisation now allows a variety of small dedicated devices—information appliances—for which the hardware, software and display is all part of the interface. Each different appliance needs to be designed to fit a specific purpose and a specific set of users.

Commoditisation: as technologies stabilise and the offerings of different companies become more the same, design is one of the things that differentiates products. In the choice of a car, for instance, performance is only one of the things people care about. They also think about what it looks like, what it feels like, how convenient it is, and what they want it to say about the kind of people they are.

The second set of reasons is to do with the changing computer industry: different users, different buyers.

Fifteen years ago computers were used by professionals and mostly bought by managers for other people to use. Today computer-based products are being used by people in all walks of life—and in their everyday lives. Computer technology has reached the consumer stage, part of everyday culture, like hi-fi or video recorders.

David Liddle (who led the team that produced the Star Interface at Xerox in the early 80s and is now president of the research company, Interval) talks about three stages in the development of technology[i]:

The first is the ***enthusiasts*** stage, when people will put up with almost any difficulties because they are so glad the technology exists at all. It either does something vital for their life or work or they are just excited by the very fact of the technology.

The second is the ***professional*** stage—in the case of computers, the office computer, bought by managers for other people who have no option but to use what they are given. At this stage the complexity of the technology is not seen as a problem, indeed people's skills in using it are one of the things they have to sell. It is not in their interests to have the technology made easier to use.

The third stage is the ***consumer*** stage where people buy things for themselves. They are not really interested in the technology, just in what it will do for them. This is the stage we are at with computer technology—it is becoming part of everyday consumer culture.

So now the problem is not just to design something that will do a job but to design something people will like and want to buy. And this is not an engineering problem—you can't engineer desire—it is a matter of style, fashion, culture, identity as well as need and efficiency.

The economy of signs

Today we live in a very different world from the manufacturing age of the nineteenth century. We live in what the French philosopher Baudrillard[ii] has called an *economy of signs*, as well as an economy of material things. Indeed, a recent report on what are called the 'culture' industries in Britain—art, music, design, film—discovered that these are now bigger than our steel industry.

We pay for things that have *meaning* for us as well as usefulness. For instance, we buy things for what they say about us as well as what they do for us: people buy Nike trainers rather than Woolworths, not necessarily because they are better, or better value, but because of the *brand*—the name and the advertising that goes with it. Indeed some things, like music or films, have no *use* at all, but we buy them just the same. So design is now about designing the meaning of things, their rhetoric, as well as the function and use of things. And these meanings shift and change as everyday culture changes.

Successful information technology products are those that recognise and respond to the culture of everyday life—people's likes and dislikes, hopes and desires—as well as fit well with the practical needs of the way they live and work. At present there are few that do either of these well.

For ordinary users the interface *is* the product. When people use Microsoft Word they have no idea if it is well engineered or not; providing it doesn't crash they are oblivious to the quality of its engineering. They are concerned with what the program feels like, how it is organised, whether it is useful to them, how elegant and satisfying it is to use. Clearly excellent software engineering is *necessary* to a good product but it is not *sufficient*; and the converse is also true: excellent design of what the system does and how this is conveyed to users is nothing without robust and elegant engineering. To draw a parallel with architecture, you can have a building that is solid and keeps the weather out but people hate living in it; similarly if you have a building that is elegant and beautiful but the roof leaks or you can't keep the temperature comfortable, it is just as much a failure. Whether we are artist-designers or engineer-designers, we need to care about both aspects for if we do not, however well we have done our own part, the result will be a lemon.

What can interaction designers contribute?

Being an artist-designers is a craft with a training long beyond formal education, closer to that of musicians than mechanical engineers. The many elements that have to be juggled in a design, from the highly practical to the purely subjective mean that there can be no perfect solution, only a range of alternative tradeoffs. This is not to say that there are no criteria for judging the success of a design, but that they in themselves do not generate a design. It appears that experienced designers, through constant practice, learn to internalise ways of optimising these tradeoffs.

Experienced artist-designers bring three main skills to the design of interactive products and systems:

Fertile imagination and the ability to think laterally: both about people, their needs and desires, and possible ways of doing things;

Visual skills: both for envisioning possible products, and imagining ways of representing systems to users;

Skill in crafting beautiful things—both elegant function as well as beautiful form and experience.

Interaction design is the design of interactive products and systems, not just the user's interaction with them—just as an architect is responsible for designing the way the overall form and function of a building fits its purpose, as well as what its facade looks like. But in the computer world people often think of designers as the people in red shoelaces who do the pretty bits at the end. Which, indeed, is one of the things they do. But if designers are only brought in at the end of the development process, this is the *only* thing they can do. Their contribution is inevitably limited to the surface elements of the interface, rather than the structure of the information and the navigation through the system. If these are not considered until after the software architecture is in place, the optimum structures from the information and user point of view may by then be impossible to implement.

So interaction design needs to consider four elements:

Function: designing what the program does and how it fits with people's goals

Representation: designing a representation of the system that allows people to understand what the system is and to conceive a model of how it operates

Values: considering how the system fits with, or changes, the values of the people who will use it and the society they live in

Qualities of experience: designing what it feels like to use the system, both at the perceptual level and the aesthetic and intellectual level.

Designing function

A project the CRD Research Studio did with LG Electronics in 1996, using their LCD display technology, is an example of a project focused on the design of function. The proposal was to imagine LCD displays dropped in price to the extent that price was no longer an issue. The designers investigated, through design proposals and prototypes, how it might change the domestic landscape.

One of the projects, by Durrell Bishop & Michael Field, proposed a universal hand-held remote-control with colour LCD screen. It is intelligent, so knows which product it is pointing at at any time, which activates the control software for that particular product.

This approach allows the designer to separate object and interface and to decide how much of the control stays with the object and how much migrates to the remote. So, for instance, the CD player has just three buttons: forward, back and eject. Anything more complex is handled on the remote, where the graphic display allows a richer and more explanatory interface. It also allows the vendor to update and change functionality purely in software.

The radio demonstrated what this might mean for the design of the objects themselves, which could now be made of a much wider range of materials. The radio is like a wooden pencil box with a slider—moving it up and down changes the pre-set stations; changing the settings and other programming is done on the remote. The station names are simply written in blackboard chalk on the slider—no black plastic, no shiny buttons.

So here the main focus of the design was on function: what would the system allow people to do and how could it be done in a different way with new technologies; the prototypes concentrated purely on explaining the functional concept.

Representation

One of the traditional skills of designers is representing products or messages to users—from toasters to posters. Graphic design, for example, gives form to ideas that people want to communicate, in ways that hold readers' attention and lead them through the content in an orderly and communicative way. Three-dimensional form gives clues to people about what a product is, and how to use it: which way is up, how to open it, turn it on and so on.

In the case of a radio for instance, its form in the past had a necessary relation to what needed to happen inside. A dial was like it was because inside something needed to go click, click, click. But computer systems are not like this: looking inside a chip won't help you to understand what the system does, it is the representation of the system, through graphics, three-dimensional form and sound, that allows people to understand it and use it.

Representations of computer systems have become steadily richer and more complex, particularly as computers began to be used to manipulate things other than numbers. Initially systems were represented typographically and graphically but with the development of sound capabilities and of ubiquitous computing, sound and 3-D form are also mediums for representation.

The desktop metaphor graphical user interface is the best-known visual representation of a computer system. But this is not the only way of representing information graphically: different kinds of representations could allow people a better handle on particular aspects of their work.

Miniaturisation of components, wired and wireless networks, and sensor technology, allow computing to migrate from the desktop into objects in the world around us—which may not have any visual display. This may be convenient in some ways, but it makes representing the system far more difficult. It needs a very different approach, building on what people know about the way the world of objects work—gravity, texture, mass shape—to allow people to build an understanding of what the system is and how they can operate it. Another project by Durrell Bishop, in which marbles represented messages in a telephone answering machine, began to explore a language of objects and to test the possibilities and pitfalls of this approach.

Humans are capable of very precise auditory discrimination: they can identify the source of the minutest sounds around their house—the creak of the third stair, the hum of the dishwasher of the people upstairs, the sound of someone opening the refrigerator, and so on. So sound can be used at the interface to make sonic icons—signs that appear like the thing in the real world, just as graphics are used for visual icons. William Gaver, as an intern at Apple computer, developed the 'Sonic Finder', a complete auditory representation of the actions of the desktop. So throwing a folder into the trash produced a sound related to its size: a deep sound for a big folder, a high note for a small one; selecting an icon produced a different sound according to whether you had selected a folder, a file or an application.

The sonic finder acted as a reinforcement of the visual representation. Sound has the useful quality that does not need our full attention to process it. Another use for sound at the interface is to allow users to keep track of background processes, like searches for instance, where users want to know how it is getting on, but not to track it all the time.

Values

Interaction design needs also to take into account human values: considering how these technology products fit in with people's personal, social and aesthetic values. Artist-designers have always operated in this realm, whether designing things as long-lasting as architecture, or as ephemeral as fashion or media. As information technologies mature and stabilise these aspects of design become more important.

There are two aspects of values to consider: aesthetic values and social values. Though in some ways they can be thought of as part of a single cultural value system, when designing it helps to think of them separately.

Anthony Dunne is exploring what he calls 'value fictions' (as opposed to science-fictions). Whereas science-fiction is looking at possible new technologies, value fictions are about what the effects of these technologies might be on personal and social values. What could a new aesthetic of the 'electrosphere' be like? It is notoriously difficult to imagine the social by-products of new technologies. We all know the story of the telephone—'every city should have one' was one response to its invention. At that time, who could have imagined today's teenagers walking down the street with their cell-phones? And as Neil Postman once asked, if we knew then what we know now, would we have embraced the automobile so enthusiastically?

Dunne has been designing scenarios to provoke us to imagine different kinds of ways we might live with technology. Taking models and prototypes as 'props', he asks people to imagine living with them, trying not only to design from his own imagination but drawing also on ideas and imaginings of very different kinds of people.

Qualities of experience

The fourth aspect of interaction design is the one most immediately obvious because it deals with the surface qualities of interactive systems: exactly how big that icon is, exactly which blue, how springy the buttons are, how lethargic the cursor feels, the insistent tone of that sound, and so on.

Many of these qualities are just those that industrial and graphic designers know how to craft so well from their experience in other media: qualities of texture, colour, balance, contrast, harmony, dissonance, surprise, rhythm. But interactive systems also have a new type of sensual quality—the kinesthetic quality of the way a system responds to you as you interact with it. This is quite new and particular to the medium of interactivity.

Understanding the iceberg

I once asked a young industrial design graduate who had started his career as an engineering apprentice what was the most important thing he had learnt. He replied that before he started he thought things looked the way they did because that was just the way they turned out. Now he knew it was because people thought it mattered and spent time, energy and care deciding just how things should be—and that people did notice, even if they did not realise it.

Much of the work of software engineers is hidden from those who have never done it themselves, and similarly the part of interaction designers' work that shows is only the tip of the iceberg. If we are to build software that works well, and works well for people, we need to understand each other's field, not so we can do each other's job, but so we can appreciate each other's skills and preoccupations.

---

[i] Liddle, David, chapter in Winograd, Terry: *Bringing Design to Sofware,* New York 1996, ACM/Addison Wesley

[ii] Baudrillard, Jean, Charles Levin(translator): *Towards a Political Economy of the Sign* , New York 1981, Telos Press

DISCUSSION

Rapporteur: Steve McGough

Lecture One

During the section where Professor Crampton Smith was stressing her belief that "the interface is the product", a member of the audience expressed a concern that this statement implied a lot more than it seemed to. The question was asked if this implied that as far as the user was concerned anything that couldn't be observed by the user was irrelevant. Professor Crampton Smith agreed and responded that as they don't know how it works, they would choose a product based on its looks and ease of use.

Another member of the audience put forward the opinion that it was more the public relations and media hype that sold the product. This view was agreed with. The participant went further to imply that it was more the logo on the product and not the ease of use or looks that sold the product; relating this to what happens in the soap powder industry.

Another participant then asked if the marketing presentation was not also part of the product interface. The response was that this was part of the interface as this is what the user brings to the product, their confidence and expectations for that company.

A further question was raised as to whether the users had an idea of the interface or did they just have some model of what the interface was based upon some sort of ad-hoc approach they had used to learn about the product. The response was that if the product was designed well then it would be easier for the user to have the correct model of the system. An example was taken of Microsoft Word where it was felt that the interface seemed to be merely a collection of functions in lists, where there doesn't appear to be a strong underlying model, except the underlying idea of the Wimp interface.

Another participant continued the above example by remarking that for Word there was no useful model of the underlying interface for a document, for example with respect to frames and graphics and the relative placement of these. That this model is not properly articulated, but is it part of the interface. Professor Crampton Smith responded to this by stating that the design of the presentation needs to be correct, and that this may be independent of the way it is actually implemented. This was then followed up by a statement about research that had shown that if the model is easily graspable and consistent people could reason about such programs much more easily.

A question was asked as to whether the mental model the user brings to the product is an object of the design. Further is it a legitimate objective to have some model that the user could understand, that was independent of the product. The response was that in the ideal world you would have generic models for things. In a way this is already here in that for a word processor people have an idea how it works, the same is also true for the wimp interface. A further question was raised asking if these situations had been designed or did it just happen. The response was that it must have been designed over months or years.

During the presentation of bringing beauty to the designs, it was asked if this wasn't the same as choreography. The response was yes but more so for a situation like this where the interaction is not as static.

During the presentation of the answer phone another person wished to know if the design was meant to be displayed or was it for eventual production into a 3 dimensional object. The reply was that it had in fact been constructed as a real object. The comments were then raised that this design was flawed in terms of safety, as a small child could eat the balls.

A second person then asked why a design was used that gave a false impression of reality. In that the user would assume that the balls would actually contain the messages when in fact the messages were stored in the answer unit and the balls triggered off the messages, giving an example of a user taking balls from his home and trying to play them on a friend's unit. The reply pointed out that although the impression of the model was false this was not one of the main points of the experiment. The main point had been to determine the effect of representing something in 3D rather than in 2D.

Another participant asked why these things were being presented, considering the danger and poor model representation behind them. He went further to state that if these were designs done by his own students he'd fail them, not present their work at a conference. Several other members of the audience then came to the speaker's defense, bringing up such issues as it is often needed for people to think about things in other ways as opposed to engineers who need to think about things in terms of safety and efficiency. Harry Whitfield then suggested that the balls themselves could be equipped with memory thus allowing the model to be consistent, and also increasing the size of the balls so that children couldn't swallow them. Another participant suggested the balls could be thought of as object identifiers as in such languages as Cobra where messages are only retrieved when a token is used.

Professor Crampton Smith pointed out that the intention of the answer phone system was never designed for actual production but merely a student wanting to experiment with what could be done with designing in three dimensions.

The audience was presented with a design for indicating the arrival of e-mail, to this the question was raised as how to determine the suitability of a product for a generation that hasn't been born yet. The advent of the home based computer and how it would have been so difficult to predict this thirty years ago was sighted as an example. The response to this was that you needed to try out weird ideas now and then assess how you think they would fit into the future.

At this point the presentation finished, although the discussion of the issues raised here continued well into the following general discussion session.