

Design Probes for Handheld and Ubiquitous Computing

Harold Thimbleby

Middlesex University, LONDON
harold@mdx.ac.uk
<http://www.cs.mdx.ac.uk/harold>

Abstract. In developing electronic gadgets, meters and oscilloscopes are used to probe into their workings, so that engineers can understand how they are behaving and how they can be improved. Analogously, this paper suggests a variety of conceptual probes that can be used to explore the process of design. The probes focus designers' attention onto successfully managing complexity and in coping with socio-economic influences on design practice in the real world.

1 Introduction

We may take it as self-evident that design should be good, yet it is also clear that it is difficult to make it so because technology is complex and because of pervasive and persuasive socio-economic forces that obscure a straight forward notion of “good.” Regardless of the abstract arguments, there is additionally a wealth of circumstantial evidence that design fails: many systems are awkward to use. There is a large literature discussing design failure [1–4], as well as business literature on the way inadequate designs become and remain dominant [5]. (We discuss the specific example of consumer devices in Section 6.)

Thus there is a tension between economically-driven complexity and an ideal of usability, elegance and simplicity. Designers and users are part of the same socio-economic system, and it is hard (if not paradoxical) for designers to stand back and be aware of the influence of subtle cultural factors. Thus various forms of explicit design inspection are advisable—asking what you have done, why you have done it, and whether the results achieve the intended goals. This paper provides a preliminary selection of *design probes*, questions that help initiate, motivate and assess design inspections.

Furthermore, if design is to be good this begs what we mean and intend by “good” itself: this question is, by definition, an ethical one. There are issues to do with awareness and control of social consequences. Design is complex, embedded in society, and there are tacit goals that may silently conflict with good intentions—users have complex preferences, partly conditioned by design conventions. They may not want “better” design. Some of our later probes, therefore, home into ethical issues, of the designers' reflection on and commitment to an ethical framework and values.

1.1 Social forces on design

Design is difficult. Economics drives design complexity upwards. Complexity suits business, for psychological reasons including the manipulation of consumer demand. In other words, unless designers have high ideals, there is no standard by which to manage complexity—it is more profitable to exploit it.

Users are at a disadvantage compared to the designers. The designers know some of the system's structure, so they are able to compress its representation more effectively than users, and hence understand it more deeply. Moreover, a user—say a purchaser of a television in a street shop—has only a few minutes to experiment and explore the system, whereas the designers have an arbitrary time in the lead up to production. Let us use the word *obfuscation* for the system design that the user does not understand but which is understood by the designers. (Obfuscation has intention in it [6].)

A manufacturer that has a large market share may be tempted to increase obfuscation. Users who have grappled with one style of device will tend to buy new devices that work similarly—because this will save learning the subtleties of different devices. So they chose devices made by one manufacturer to ensure similar design. In communities, the effect of this sort of reasoning is profound. In a school community a classroom of different obfuscated products is hard for a teacher to manage, so the teacher will recommend a specific product. Naturally, this is likely to be the market leader. Thus by this strategy, the market leader obtains an entire class room as consumers of the same product.

Alternatively, if the user despairs of understanding the product, their choices are easier to manage. The user is no longer driven by rational processes based on an assessment of the quality of the product, since by assumption this is too hard to do. Instead, the consumer is guided by appearance, pricing, brand name, and so forth. All these are easier factors for a manufacturer to control systematically—for instance, by advertising, discounts, or product placement. Some manufacturers may simply promote several brands, and thereby increase the probability that consumers choose one of their brands. (If there was no obfuscation, the consumer could distinguish the brands, and not be fooled that they offered choices.)

Given these issues: design is difficult, systems are exceedingly complex, and manufacturers have no incentives to simplify. We end up with a society not using technology but driven by it—a designer who wishes to design well needs some sharp insights.

For further discussion, see [7–9].

1.2 Technical forces on design

A single line of Java code can easily generate hundreds of bytes of object code; a Java application includes packages written by other people and may end up being thousands of times larger than the Java program that specified it. What the code does is essential to understand, since this is what the device runs, and it specifies what the device does, step-by-step. Although a device may run the code directly,

it is of course much more convenient for a human to write and understand the Java, because it compresses the program represented by the code.

This is a brief argument why high level programming languages like Java are used: they compress object code (which tells machines what to do) and make it easier to get the designer' minds around. Java is a successful language because it compresses useful programs more efficiently than many other languages, such as C or Cobol (though we tend to think of programs being expanded *from* Java, rather than Java representing the object code). Thus, programmers can produce more sophisticated programs with the same effort, or reproduce existing programs with less effort than before.

Now the nature of competitive computing business (brought about by ubiquitous computing) is that the market will force programmers to operate at the limit of their capabilities: for if they did not, some other programmers could produce cheaper, more reliable or more sophisticated systems at lower prices. The improved compression of Java does not make programming easier, at least under commercial pressure, it raises the threshold—programs are always designed at the limit of the programmers' competencies.

Unfortunately, the compression of object code is not sufficient; there is more to design than program text. Programs have dynamic behaviour that unfolds as they interact, and this is harder to compress than their static description in code. Some programs, of course, have trivial behaviour that is easy to compress and represent concisely. Such programs' behaviour is easy to understand; unfortunately, any program text that is easy to understand is likely to represent a program that is not very interesting to use.

Consider a program that interacts with a user, giving the user a modest ten choices every minute (for instance, from a menu or a selection of push buttons). At the end of just a quarter of an hour, the user will have selected one out of 1,000,000,000,000,000 possible outcomes. With more choices, making them faster, or spending longer interacting, the number of possible outcomes increases astronomically. It is clear that the designer cannot anticipate, except in the broadest of terms, what a user will do.

We now provide a concrete example to show how daunting complexity gets even in the most mundane applications. A simple digital alarm clock can display 1440 times (00:00 to 23:59), and the alarm can be set or not set to any of those times. At any time of day or night, the user may set the alarm clock for any time in the next day, so there are 2,075,040 (including the case of not setting the alarm) possibilities that the designer has to be sure work correctly. To check each possibility, the clock has to be run for a whole day to check the alarm sounds once at the right time (or not at all if it hasn't been set).

Doing the checking non-stop but exhaustively would take 5,685 years. Nobody—in fact, few civilisations!—can wait so long, so various cunning tricks are used instead of an exhaustive check. In practice, statistical quality control techniques would be used, but there are more analytic approaches that even the statistical methods assume ...

If we checked 1,000 alarm clocks *at the same time*, we could get the total checking time down to six years. Better still the designers should arrange the clocks so that they were sure that the clocks had symmetries—for instance, **if** they work between 1 and 2 o'clock **then** they will work between 2 and 3 o'clock, and so on. In fact, checking 1,000 alarm clocks assumes a symmetry, namely that when we make two alarm clocks they work the same way—they accurately “reflect” each other’s design. By using symmetries, the task of checking can be reduced to manageable proportions. Fortunately many symmetries for alarm clocks are easy to conceive.

2 Symmetry

In everyday language, symmetry refers to mirror and rotational symmetries. Thus when a geometric object can be reflected or rotated and appear unchanged it is said to be symmetric. For example, a circle has both symmetries: reflecting or rotating a circle leaves it coincident with its former self. But further, a straight line can be moved along its length (i.e., translated) and left coincident with its former self. Thus translation is a symmetry. Translation through time is an important symmetry: if a system behaves like this today, it will behave the same way tomorrow. The notorious Year 2000 problem is a break in time symmetry, and an example of the seriousness of breaking symmetries people rely on to understand and use systems.

If the alarm clock (mentioned at the end of the previous section) was a mechanical one, with gears, the symmetries would be familiar ones: gear wheels that have regularly spaced teeth have simple rotational symmetry. Once you are sure a few teeth work, you can be convinced by symmetry that the others will. Other geometrical symmetries are important for checking: for instance, symmetry under translation ensures that the gear wheels still work if the whole clock is moved. Symmetry is a more general concept, however, and can be applied to non-physical structures, such as programs, that are not round and not even geometrical.

Humans are familiar with physical objects, with physical symmetries. The symmetries applicable to more sophisticated but less physical systems, such as video recorders or avionics systems, are harder to conceptualise or work out in the first place; quite possibly the necessary symmetries will be found less effective in compressing the descriptions of the systems. It follows that complex systems do not have convenient symmetries. Users—as well as programmers—will find them hard to understand.

Since designers are not sure what symmetries to require, and because computers are such flexible systems with large memory capacities, it is easy to introduce features to designs without any regard for the effect on the complexity of the system. Put in other words, systems often have large numbers of unrelated features, because it is easy to design them this way. Coherent features require a lot of hard design work, and anyway users can easily be sold features rather than comprehensibility. If computer memory was a restricted resource, then the

program would have to be more carefully designed—typically by using a small number of carefully-designed general functions that can be applied in many parts of the program [10]. If the structure of these shared functions is visible in the user interface, then the system will have symmetries. For example, all scroll bars in a graphical user interface have rules that should be symmetric: changing scroll bar should not change the properties. However in a badly designed system, some scroll bars (or other features) may have been programmed specially, and therefore have some arbitrary properties.

Graphical user interfaces have become popular because a set of powerful symmetries are readily defined and agreed. Indeed they are inspired by familiar geometrical symmetries: windows and scroll bars work the same way under translation and scaling. The problem with push button user interfaces is that there are no very obvious symmetries; one might say that in the course of evolutionary time scales, push button-controlled complex gadgets have not had much impact on natural selection, and therefore humans are not selected for understanding them. Rather than wait many generations to experience selection forces (i.e., disasters), rather than wait for the lessons of numerous accidents to be widely learnt and intuitively appreciated by society, we need some more powerful methods for design. Until then, cultural pressures—such as those we described on the one hand forcing programmers to exceed their competencies, and on the other hand forcing users into a culture of complacent idolisation of technology (Section 1.1)—will drive design to be worse.

Probe 1. *What properties does the design have that work in more than one place? Are there symmetries across the system (i.e., does it have a coherent design); are there symmetries from one system to another (i.e., does it provide useful communications properties)?* ◁

Probe 2. *Are there properties shared with other material, such as user manuals, design documents . . . ? Are there symmetries that work across media—the device, the manual, the user’s head?* ◁

For further discussion, see [11, 12].

• • •

This paper continues with this structure: discussion of design issues, including a brief listing of probes. A discussion of the nature of probes, of the answers to probes, and how probes differ from design heuristics, is postponed to Section 10.

3 Drama

You are watching a film, a play or a TV programme. The story is engrossing, and your mind is following the story as it unfolds. What will happen next? Who will do what? These are real questions to the audience, yet in reality the play is running from a script and there are no choices. If the medium is a film, the reel has no junctions in it: it will unwind the one story it was made for. In reality, there are no choices, and everything is trivial! But that is not how it

seems. The story may be exciting, frightening, worrying, or romantic. Our minds automatically create a wider world, and our feelings operate as if the media was reality. Of course that is the point, and a good play or film works its magic carefully to achieve its emotional impact. The drama need not be Shakespeare to work so deeply; it might be a simple demonstration of an interactive product. A story unfolds about how the product is to be used, demonstrating this feature and then that feature. Because our minds work the same way as before, we envisage a grander plan that we emotionally suppose the demonstration is but an example of, or a story within. But often a demonstration is not only just a linear story like a film reel, it was only designed as such.

A demonstration, then, can create the impression of a more sophisticated system than may actually exist. We know that Macbeth is fiction and we know some questions are just not worth asking—but someone watching a system demonstration does not know what is real and what is merely demonstrated from a script. The demonstration may even be stage-managed by a human, who works through a script behaving exactly as if they were using a fully working system. But there is no way to tell, merely by watching the scripted demonstration, whether there is any more to the system than is shown. Nevertheless, we imagine that there is, and we imagine that it all works as smoothly as the fraction we saw demonstrated. Reeves and Nass [13] argue that this is how our brains work: media is too new an invention for our emotions to have taken it on board. Evolutionarily speaking, up until very recently all experiences we had were real; only in the last few decades (with the exception of theatre) have we had media where reality may or may not be faked. Hence Reeves and Nass define the *media equation*: media equals reality, so far as we are subliminally concerned.

The media industry exploits this effect and we enjoy it. But in systems design, there are serious issues:

- A designer may demonstrate a system and marketers will become too excited by the potential of the design that they insist the product is ready. In fact, the product may be no more than a linear simulation.
- A user in a showroom may be shown a carefully rehearsed demonstration that makes the device seem much more effective than it really is (for that user in particular).
- A scientific paper may describe how a system works—and may deliberately or accidentally not mention that the paper is “but a stage,” with edges. Beyond the edge of the stage, the media breaks down; the reader of the paper may imagine the system is more powerful than actually described. But the paper (even if completely truthful) may not define the system’s general behaviour. The reader naturally assumes the paper describes a system with symmetries; if those imagined symmetries are not true in reality, the reader has a more powerful image than the writer of the paper had.

The purpose of drama, if that is what it is, is to stimulate, not to exaggerate. When people present papers or describe systems (possibly demonstrating systems) ask,

Probe 3. *Is the design as presented real or drama? ◁*

The problem with drama is that it takes a tiny view of a large, complex design. The complex design that the viewer imagines may not be the actual design. Unfortunate unwarranted generalisations from drama can be avoided by making design principles explicit: then any drama becomes an illustration of a principle the design satisfies.

Probe 4. *Is the drama backed-up by explicit principles that it illustrates? In other words, is it made out to be an illustrative instance of a larger class of possible demonstrations, or is it a unique experience that illustrates little of general value? ◁*

For further discussion, see [14, 15].

4 Explanation

The Canon EOS500 has been one of the most popular automatic SLR (single lens reflex) cameras. The EOS500 manual [16] warns users that leaving the camera switched on is a problem. Canon evidently *know* that the lack of an automatic switch-off is a problem: there is an explicit warning in the manual on page 10:

“When the camera is not in use, please set the command dial to ‘L.’ When the camera is placed in a bag, this prevents the possibility of objects hitting the shutter button, continually activating the shutter and draining the battery.”

So Canon knows about the problem, and they ask the user to set the camera off—rather than designing it so that it switches itself off. Thus Canon is aware of design problems, but somehow fails to improve. The user manual for the EOS500N [17], an improved version of the EOS500, puts the same problem thus:

“If the film is removed from the camera in midroll without being rewound and then a new roll of film is loaded, the new roll (film leader) will only be rewound into the cartridge. To prevent this, close the camera back and press the shutter button completely before loading a new roll of film.”

It seems the manual writers have now discovered that as well as pressing the shutter button, the camera back must be shut too (it would probably be open if you were changing a film). But it does not seem like the camera designers read the EOS500’s manual themselves. User manuals are obvious forms of explanation, to explain the design of a product to its users. Evidently if designers read user manuals (and were able to act on what they learnt) designs might improve.

Probe 5. *Has a complete and truthful user manual been written, and what warnings did it make? Could the design be improved so that the warnings become unnecessary? ◁*

Probe 6. *Are the training procedures complete and truthful, and what warnings do they make? Could the design be improved so that the warnings become unnecessary?* ◁

Since having many views of the design that are complete and sound is essential, then we have higher-level probe,

Probe 7. *Are automatic methods used to guarantee all views, explanations, training material, documentation, and the design itself are complete, sound and consistent with each other?* ◁

For further discussion, see [18, 19].

5 Lottery effect

One of the best ways of handling complexity is to ignore it. In computing, we can write programs and never bother to try to understand them: thus, features are added, and the overall coherence of a design is never addressed. Of all features a sufficiently complex system exhibits, some will, just by chance, be rather effective. If we concentrate on those features, the system as a whole looks effective. The point is that we can find the effective features after an *ad hoc* design process, and in hindsight we can make the design look better than it really is.

Numerous organisations (and some countries) run lotteries. The media bombards us with the success stories: each week somebody has won lots of money, and this is news reported everywhere. Thus we become familiar with winning. As the media equation argues, we are not used to this mediation and we assume that if we are familiar with something then it must be probable. It is but a short step to think that as lots of people win the lottery, we are likely to win too. Of course, if newspapers reported each week the millions of disappointed people who failed to win, thus putting the one winner's chances into perspective, then we would have a very different (more realistic!) view of the lottery.¹

In product design, we create a complex system and then describe part of it—in a paper at a conference, in a demonstration in a store, in a presentation to the marketing department—and the people who watch the demonstration are in danger of suffering from the lottery effect. If we emphasise success, for there is surely some success somewhere in the design, then the demonstration exploits the media equation, and gives the viewer a biased view of the design.

The lottery effect has a further twist. Many systems involve communications; anything on the Internet or using radio communications necessarily interacts with other people. Suppose such a system fails—for example, a web-based business fails—then nobody gets to hear about it. In fact, in a communications medium, people only hear from successful designs, and do not hear from unsuccessful designs (unless they are sufficiently notorious to be reported in the

¹ You are more likely to win in the UK National Lottery by betting at the end of a week—if you bet earlier, you are more likely to die than win.

media). Unsuccessful designs do not communicate, and they disappear. Thus not only is there the lottery effect, but there are actually “no failures” to report!

Probe 8. *From the successful examples, how successful can we say the whole system is? If the examples were selected randomly, then their success suggests the whole system is good; if however the examples are selected with a bias, then we do not know much about the overall design. What method was used to select the design scenarios? <*

6 Blame incompetence

A very effective way of coping with unmanageable design complexity is to suggest that it is someone else’s fault, or to find other reasons why it cannot be our fault. Better still, define social culture so that the scapegoats willingly accept their role.

The video recorder makes a classic example of this effect. *(i)* People find that their children can use a video recorder better than they can, and they conclude that they are getting too old. *(ii)* People find the manuals incomprehensible, and the technical authors (usually people in a foreign country who evidently don’t speak the relevant language fluently) become the scapegoat. *(iii)* There are new models recently introduced on the market that solve the problems; if there are easier products on the market, the user must have made a mistake buying their current gadget. In short, the user is failing in their duty as a consumer to buy the newer, easier-to-use products. *(iv)* We haven’t yet got books called *VCRs for Dummies*—as there are for computer programs—but the attitude, from all directions, is that users *are* dummies, and it is their fault [20]. *(v)* After any major disaster, the explanation is often human error, user error, pilot error, or operator error. All are familiar terms. The law, at least in the United Kingdom, requires individuals to take the liability for accidents. It follows that users are responsible.

All of these attitudes can be challenged:

(i) Children do not know how a video recorder ought to work, so they press buttons almost at random. This behaviour is likely to generate examples of what they want to achieve, and they can then generalise their discoveries to do what they want. In contrast, adults—having learnt how the world works—expect video recorders to work in a particular way. Since they do not, adults are forever frustrated. Furthermore, children are not responsible—they have not just spent lots of money on the gadget, and they are not worried about “breaking” it by using it the “wrong” way. In short, children are more successful at using video recorders, but not because adults are old and “past it” but because video recorders are badly designed for mature people to use [21].

(ii) Manuals are incomprehensible because a bad design has no good explanation, in any language. (This point is elevated into a probe in the next section, below.)

(iii) There are always new models being introduced on the market. Marketing for last year’s models, at the time, presented the objects as desirable and solving

the previous year's models. That there is always a solution to bad design (i.e., buying another design whose besetting problems will not become apparent to the user until a much later than the point of purchase) is not an excuse for bad design.

(iv) In the 1960s Ralph Nader intercepted a memo between two companies, asking how drivers could be trained to park cars better [22]. On a hill, cars not parked properly tend to roll down the hill. If parked cars roll down hill, then the driver needs training to do a better job: thereby the issue is presented as one of driver training, driver psychology, and even of writing good driving manuals. In fact even for reasonable inclines, the cars were hard to park on hills because the parking brake was flimsily engineered. Thus an engineering fault was being passed off as a user fault. Clearly *some* errors are user errors, but many are design errors passed off as user (or driver) errors.

(v) When there are substantial investments in hardware, as in avionics, it is tempting to blame "pilot error" since the pilot is unique, and if the pilot has made an error then nothing need be done to the expensive hardware. In contrast, if the airframe is blamed, there is a danger that an entire fleet of aircraft have to be grounded to fix the problem. Since pilots often die in accidents, there is often no strong argument that they are not to blame. Because of the cost of rectifying design errors in contrast to the cost of retraining (which is zero if the pilot is dead), there is an overwhelming tendency, regardless of the facts, to blame the human most closely involved at the moment of error—rather than the designer.

Finally note how one's difficulty with using a video recorder is celebrated as a joke: it is now difficult to treat the subject seriously. A satire of *Star Trek* had one of the security officers fumbling. The captain asks why the officer is not ready to go, and the officer replies that they cannot get their phasor to stop flashing 12:00! ... Nobody can get their video recorders to stop flashing 12:00, and the joke is that the officer's phasor has become as complex as a video recorder and now defeats its very purpose—namely to be a ready-to-hand tool.

If an interactive system does not work, there is a mismatch between the user and the device. It is easier, sometimes, to focus on the user's problems, rather than on the design problems. Users are "more easily changed" than systems, and in a marketplace, the users fixing "their" problems will cause more product sales—whether selling how-to manuals, or upgraded products promising to solve the spurious problems of the obsolete products. That millions of tons of electronics are thrown away and used as landfill every year is a testament to the economic power of persuading users that it is their fault that designs do not work, and that therefore the users should invest in new solutions—hence discarding old solutions.

Probe 9. *Are design failures discussed, and if so who is blamed for them?* ◁

Probe 10. *Are people defending bad designs with spurious arguments?* ◁

7 Scientific method

Perhaps it is hard to recognise, but all of the issues described above are analogous to issues of scientific method. The purpose of science is to have reliable knowledge of the natural world, and the scientific method has developed so that humans, with their limited abilities and tendencies to make errors, can work collectively to understand and check each other's results, and reach consensus [23]. In design, which we have been discussing, the issue is to understand artificial worlds, and to increase the community of people who understand and can use the underlying ideas. A successful design is replicated, and many people end up using it, employing the same underlying theory of operation. The current boundaries of science are precisely the still difficult-to-understand areas; the current boundaries of design are precisely the still difficult-to-understand ideas. In science, curiosity drives the boundaries; in design, economics fights back. In both, anything easy to understand is less interesting.

Both science and design, then, push human limits. In science a very clear set of protocols has emerged to help people resist and detect the temptation to cheat. One of the core principles is *replication*. Is a scientific result described in enough detail for someone else to replicate it? Was selective bias (cf. the lottery effect of Section 5) avoided in reporting the idea? If some data is held back, then it might be an error that the scientist is trying to conceal—or it might, if revealed, help another scientist to find a better explanation of the result than that claimed.

In computing we have exactly the same problem, exacerbated because most design products are seen, most often, in a consumer world where competitiveness and industrial secrecy rather than openness is the norm. The consumerist comparison is powerful because many consumer gadgets—on sale in stores—are sophisticated and powerful in comparison with experimental work; designers easily envy the seduction of consumer products and therefore aspire to “worldly” rather than “scientific” standards.

One difference is that a good design has expressible symmetries. These are the ideas that other people can understand (because they are expressed) and use in their own work (because they are symmetries). A bad design has inexpressible properties that are merely attractive, it has boundaries that break symmetries—in short, it works only as drama. The idea it represents is no bigger than the stage it is seen on.

Student projects in computer science make a good illustration of the tension. So many students work hard to build a thing. The thing is interesting in itself, but what remains of the student's work in a year's time? The system probably no longer works, because it relies on obsolete computers. All the project was, was a dramatic performance—despite containing a program that in principle could be replicated and run anywhere anytime, it was a program that merely created a time- and space-limited experience.

Probe 11. *Have the design ideas been expressed clearly and fully-enough for anyone else to replicate the core ideas? If not, what exactly has the design contributed to the future? <*

8 Ethical commitment

William Perry claims that we learn and understand subjects on a nine-stage scale of increasing intellectual sophistication [24]. Designers work at various positions within Perry’s scheme. The initial position is that the designer believes in a basic duality: there are right and wrong answers. As they become more experienced, they realise that some authorities disagree—but they still cling to the idea that there are right and wrong answers, so that some of these authorities must be wrong. At least they are learning that their subject is hard!

At higher positions of intellectual development, designers realise that what is right or wrong depends on their commitment; there are different styles and approaches—they need to make a choice of what they want to be. Next—and this is a significant leap—they *make* a commitment. Perhaps they want to design better for the under privileged, or for the rich? Next, the last position, there arrives a higher level of intellectual sophistication: that the commitment itself is dynamic, contextual, and indeed a life’s work.

Probe 12. *Is the design presented as right (as opposed to other designs that are wrong), or is a more mature view presented? Is the designer’s commitment to the design orientation explicit, or not yet worked out? <*

In this paper we claimed that symmetry underlies good design, and that social, cultural, cognitive, evolutionary and economic forces conspired to make it hard to see clearly. We discussed probes, analytical tools to help designers climb out of the fog of complexity, to avoid the Siren songs of image over integrity. Nevertheless, we kept one factor implicit: the definition of good itself. Until a designer knows and chooses what sort of good they are pursuing, they will run the risk of being misled—of making arbitrary decisions. In an informal sense, ethics is the highest symmetry; a designer’s ethics is that property that infuses all their designs, and is invariant from their mind’s conception right through to the deployment, use and impact of the artefact in the hands of its users. Moreover having an explicit ethic gives the designer an integrity that they are not easily swayed from: each day, as it were, the designer reviews their behaviour and performance against their ethic and aims to avoid pitfalls next time. Only by having an ethic of which they are not ignorant have they a firm place from which they can see, and perhaps even anticipate, pitfalls. In short, designers need symmetries: properties that remain invariant from design to design.

One might derive a heart-warming feeling from being engaged in design and knowing design is doing good—a reassuring but hardly insightful thought. To be more specific, a characteristic of design (rather than art) is that there is an enormous scale-up (e.g., by mass production or software copying) from designer to user community; since a designer “lets go” of a product, for most people the

relevant ethics involve the balance between the designer’s greater influence as against the greater number of users.

Interestingly, we could make an analogy between designers of complex products—who define rules for the product’s operation—and actual rulers, in the classical sense, of tyrants—who define rules for a society’s operation. Indeed one might set up a translation of classical ethics from society and map it into design. There are good rulers and bad rulers; there are good designers and bad designers. Aristotle defines *justice* as that virtue concerned with doing good for others [25]. If justice is the ethics of concern for designers, it is disappointing that even after almost 2,500 years the correspondence between justice and design is obscure. However, in the 1970s John Rawls introduced an operational conception of justice [26].

A just society, according to Rawls, is one which was or was in principle defined by legislatures working under a so-called *veil of ignorance*. If I know who I am to be in a future society I am designing, I may design it so that it is advantageous to me, or to people who share some of my properties (such as my level of computing skill). If, however, I am under a veil of ignorance and I do not know who I shall be in this society being designed, I shall try to design justly—because I might be anybody, with any properties. In fact, almost by definition, I would be unlikely to be a privileged member of the society, and therefore my rules would tend to give rights to large sectors of society, since statistically I am more likely to be in these sectors.

If a designer acts justly (i.e., under the veil of ignorance) they cannot design for themselves; they must “know the user”—a standard design slogan, at least since Wilfred Hansen’s classic 1971 article on design [27]—for if they do not they are creating a system into which they could find themselves with the “wrong” properties. For example, acting ignorantly they might design a system that requires hand-eye co-ordination; acting knowingly under the veil of ignorance, they would check whether their users had such co-ordination. Clearly, if their users are an ageing population, or car drivers who have to concentrate on the road rather than the in-car entertainment system [28], then knowingly acting justly will result in a more effective system.

Probe 13. *What ethics does the design represent? In particular, is the activity just; does it do good for others? ◁*

Just design implies providing benefits to users (often balancing difficult trade-offs); but, as argued in Section 1.1, users have *counteradaptive preferences* [29]; clearly, design for users *with* such preferences (especially when they are unreflective) is different from innovative design stimulating rational choice.

Probe 14. *Does design make user benefits explicit (or does it substitute fashion or consumerism)? ◁*

Probe 15. *Does a design persuade users to do something the designers would not want to be persuaded to do themselves? ◁*

For further discussion of justice applied to design, see [30] (from where the last probe was taken) and [31].

9 Probing towards a global ethic

Design has been described so far as an almost private affair between the design and a group of people, the users. In fact users are far from homogeneous, and with international communications, design has become a global, multi-cultural issue. The global world of users are unlikely to agree on a common ethical basis. It is best, then, to make the ethical probes behind the design as explicit as possible.

Hans Küng in his *Global Responsibility: In Search of a New World Ethic* [32] proposes six rules for priority and certainty (which he attributes to Dietmar Mieth). Here, we have converted them into probing questions for designers.

Probe 16. *What problems are solved by the design?* ◁

There must be no technological progress that creates more problems than solutions. This probe requires us to think through the consequences of a new idea. It is perhaps too easy to think only of the advantages to some people in some conditions when promoting a new technology—this probe requires us to think about the problems too. Technology often causes problems; we must try to find designs whose advantages out-weigh their problems (so far as we can foresee them).

An important point this probe raises is the assumption that technology works as intended. Almost always, technology will solve problems. But what if things don't work out the way that was intended? What if the technology is less reliable, or if criminals find ways of subverting it? What if we have another problem on the scale of the Year 2000 bug?

Probe 17. *How do you know the problems are solved by the design?* ◁

Anyone who introduces new design must prove that it does not cause ecological, social or other damage. People make all sorts of claims about technology. How should we believe them? Have they done experiments, and what were the results? Society should require certain standards of performance, and then designers should meet those standards.

The Year 2000 bug is a nice illustration. Who, exactly, proved things would work? Why is no-one who made the faulty goods responsible for fixing them? . . . The answer is partly because no consumers ever made designers or manufacturers prove their original claims: we all took the benefits of computers on trust. This probe was not used; and now we are paying the cost.

Probe 18. *Does the design benefit a large part of society?* ◁

Interest in the common good has priority over the benefit of particular individuals, provided human rights are preserved (a comment guarding against a totalitarian view of “common use comes before personal use”). This probe of Küng's is an expression of user centred design (UCD) [3].

Probe 19. *Are more urgent issues being solved?* ◁

The more urgent (such as the survival of an individual) has priority over higher goals (such as self-fulfilment for the designer or for a particular user). This probe suggests, other things being equal, that implants like pacemakers are more important to develop and have than implants like calculators!

Probe 20. *Is the environment respected?* ◁

The ecological system has priority over the social system. The reason for this is that we can change designs, but that many changes to the environment are irreversible. It is hard to “lose” a design concept, but it is easy to lose a species, or to pollute some land. Since our children inherit the planet, then this probe suggests we think about the future, and that we should ask whether our technologies do good for our descendants.

Probe 21. *Is the impact of the design reversible?* ◁

Reversible developments should have priority over irreversible developments. For example, if we are going to have brain implants, we ought to be able to change our minds later.

Some people will not agree with what they see as implied political bias in the probes. But note that this is a list of probes; they are not rules. It may be that you are thinking about a new feature for a system design. There may be no obvious way that the idea is “reversible” or that it promotes the environment over the social system . . . well, the probes are not rules. The point is that the probes will have helped designers think about significant trade-offs, and—in particular—started to ask questions about the human worth of what is being undertaken.

As Küng himself points out, this is a rational list. But as the probes are used, the more specific the probe’s wiggling becomes, the more questions are begged about motivation, the degree of compulsion, and the ultimate issues of right and wrong.

10 The purpose of probes. What are the right answers?

Used appropriately, probes are constructive in the overall design process. However an intimidating number of probes were presented in this paper, ranging from ones concerned with the process of design, through to ones concerned with the ethical stance of the designers. Too many to be incisive! Indeed, if the probes were written down as a check-list, the reflection they are intended to inspire would not come about. There would be a serious danger of designers running down the list, ticking boxes quickly. The probes in this paper were instead intended to offer a selection from which a pertinent focus can be constructed. For example, a design company may wish to develop probes that closely match its corporate mission statement—or, conversely, it may choose a mission so that designers can operationalise it directly in their work.

Probes focus on key, but often ignored, properties of design, and help communication between designers, between designers and users and beyond. Unlike

heuristic evaluation [33], the issue is not the design itself so much as the design process. Like probes on a volt meter, design probes are to be pressed home on the relevant parts of a process—even pressed on the designer—and to help obtain a reading of the “voltage” or other property. There is no *a priori* sense in which a high voltage or a low voltage is better or worse; it depends on what the purpose is whether one extreme or the other, or indeed an intermediate value, is desirable. But once we have a conception of “voltage” we can talk about it, get clearer readings, understand each other better, and improve the design processes in the direction we wish to go.

How do probes differ from design heuristics? A heuristic tries to make suggestions that would improve *a* design; for example *equal opportunity* is a heuristic [34] that suggests that user interfaces should treat input and output equally (a symmetry) and that doing so will make interfaces simpler and more powerful. In contrast, probes do not improve design; they improve thinking. There is no right answer for a probe—the point is that not having a good answer, or not being able to recruit an answer is, the problem that a probe would highlight.

Heuristics try to be constructive; they may be analytical and provide quite specific guidance, or they may be intuitive and inspirational. In contrast, probes are critical, though (like heuristics) they may be specific or inspirational in style.

In the long run, converting probes to constructive heuristics will be a valid research exercise, but it is not obvious how to proceed along this path at the moment. (See [35, 36] for examples of placing heuristics into a design process. In particular [18] suggests that, by using computers appropriately, design can be made concurrent, and hence, by being more efficient, better things can be achieved within given resources.)

I believe probes are an improvement on approaches that do not encourage debate with the designer (such as [37, 38]). It seems to me that the idea of probes allows us to start asking penetrating questions about where the design of interactive computing systems is taking us, where we want to go, and to warn us off diversions.

11 Conclusions

Design is complex, and it occurs in a complex world whose standards are different from the ideal of pursuit of good design. Thus the practical problem of design is to have clear discernment, to perceive the forces that result in compromises or, worse, in designs that look right but are inappropriate. Designers are agents of social change, yet, being at the top of a certain sort of social pyramid (there is one designer for a thousand users), they are caught up in conventional social forces. We believe designers and hence designs must have integrity. How anyone achieves that is a hard problem; this paper contributed by espousing a range of probes. The purpose of the probes was to raise issues, to raise consciousness of alternative points of view, and ultimately to show design as a political activity of changing the world—hopefully for the better.

The properties probes reveal are a consequence of symmetries; in particular, of properties that are conserved as they are translated from one mind to another. If there are no such symmetries, design becomes art, valuable because it is unique, and diminished if it is multiplied (“forged”); design becomes repetitious drama, transiently satisfying, non-interactive, ultimately predictable, and of no lasting value.

Acknowledgements

Ann Blandford, Penny Duquenoy, Peter Ladkin, Michael Harrison and David Pullinger made very useful comments for which the author is grateful.

References

1. H. Petroski, *To Engineer is Human: The Role of Failure in Successful Design*, MacMillan, 1985.
2. D. A. Norman, *The Psychology of Everyday Things*, Basic Books, Inc., 1988.
3. T. Landauer, *The Trouble with Computers*, MIT Press, 1995.
4. H. W. Thimbleby, “You’re Right About the Cure: Don’t Do That,” *Interacting with Computers*, **2**(1), pp8–25, 1990.
5. C. M. Christensen, *The Innovator’s Dilemma*, Harvard Business School Press, 1997.
6. L. B. Slobodkin, *Simplicity & Complexity in Games of the Intellect*, Harvard University Press, 1992.
7. N. Postman, *Technopoly: The Surrender of Culture to Technology*, Vintage, 1993.
8. E. Tenner, *Why Things Bite Back: Predicting the Problems of Progress*, Fourth Estate, 1997.
9. M. Piattelli-Palmarini, *Inevitable Illusions: How Mistakes of Reason Rule Our Minds*, John Wiley & Sons, 1994.
10. R. Bornat & H. W. Thimbleby, “The Life and Times of Ded, Display Editor,” in *Cognitive Ergonomics and Human Computer Interaction*, pp225–255, in J. B. Long & A. Whitefield, editors, Cambridge University Press, 1989.
11. J. Grudin, “The Case Against User Interface Consistency,” *Communications of the ACM*, **32**(10), pp1164–1173, 1989.
12. E. Gamma, R. Helm, R. Johnson & J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley, 1995.
13. B. Reeves & C. Nass, 1996, *The Media Equation*, Cambridge University Press.
14. H. Thimbleby, “Internet, Discourse and Interaction Potential,” in L. K. Yong, L. Herman, Y. K. Leung & J. Moyes, eds., First Asia Pacific Conference on Human Computer Interaction, pp3–18, 1996.
15. B. Laurel, *Computers As Theatre*, Addison-Wesley, 1991.
16. Canon Inc., *EOS500/500QD Instructions*, part no. CT1–1102–006, 1993.
17. Canon Inc., *EOS500N/500NQD Instructions*, part no. CT1–1111–000, 1996.
18. H. W. Thimbleby, “Specification-led Design for Interface Simulation, Collecting Use-data, Interactive Help, Writing Manuals, Analysis, Comparing Alternative Designs, etc.,” *Personal Technologies*, **4**(2), pp241–254, 1999.

19. H. W. Thimbleby & P. B. Ladkin, "A Proper Explanation When You Need One," in M. A. R. Kirby, A. J. Dix & J. E. Finlay, eds., BCS Conference HCI'95, *People and Computers*, **X**, pp107–118, Cambridge University Press, 1995.
20. H. W. Thimbleby, "Minotaur," *Ariadne* (ISSN 1361–3197), **18**, p11, 1998.
21. H. W. Thimbleby, "The Frustrations of a Pushbutton World," in *1993 Encyclopædia Britannica Yearbook of Science and the Future*, pp202–219, Encyclopædia Britannica Inc., 1992.
22. R. Nader, *Unsafe at Any Speed*, Pocket Books, 1965.
23. J. Ziman, *Reliable Knowledge: An Exploration of the Grounds for Belief in Science*, Cambridge University Press, 1978.
24. W. G. Perry, Jr., *Forms of Intellectual and Ethical Development in the College Years: A Scheme*,² Jossey-Bass Publishers, 1999.
25. Aristotle, *Nicomachean Ethics*, Book **V**, in *Great Books of the Western World*, **8**, Encyclopedia Britannica, 2nd ed., 1990.
26. J. Rawls, *A Theory of Justice*, Oxford University Press, 1972.
27. W. J. Hansen, "User Engineering Principles for Interactive Systems," AFIPS Conference Proceedings, **39**, pp523–532, 1971.
28. H. Thimbleby, P. Duquenoy & G. Marsden, "Ethics and Consumer Electronics," Ethicomp'99, in press, 1999.
29. J. Elster, *Sour Grapes: Studies in the Subversion of Rationality*, Cambridge University Press, 1985.
30. D. Berdichevsky & E. Neunschwander, "Towards an Ethics of Persuasive Technology," *Communications of the ACM*, **42**(5), pp51–58, 1999.
31. H. Thimbleby & P. Duquenoy, "Justice and Design," Interact'99, in press, 1999.
32. H. Küng, *Global Responsibility: In Search of a New World Ethic*, translated by J. Bowden, SCM Press, 1997.
33. J. Nielsen, *Usability Engineering*, Academic Press, 1993.
34. H. W. Thimbleby, *User Interface Design*, Addison-Wesley, 1990.
35. H. W. Thimbleby, "The Design of a Terminal Independent Package," *Software—Practice and Experience*, **17**(15), pp351–367, 1987.
36. H. W. Thimbleby, "Designing Interfaces for Problem Solving, with Application to Hypertext and Creative Writing," *AI & Society*, **8**, pp29–44, 1994.
37. H. W. Thimbleby, "Failure in the Technical User Interface Design Process," *Computers and Graphics*, **9**(3), pp187–193, 1985.
38. H. W. Thimbleby, "User Interface Design," in *Software Engineer's Reference Handbook*, J. A. McDermid, ed., pp57/1–57/14, Butterworth-Heinemann, 1991.

² This is the Library of Congress title; the title of the book is *Forms of Ethical and Intellectual Development in the College Years: A Scheme*.