APPLYING THE PRODUCTS OF RESEARCH ON INTERACTIVE DIALOGUES

P.J. Barnard

Rapporteurs: Mr. D.R. Brownbridge
Mrs. J.A. Anyanwu

Abstract:

Apart from the problems of accumulating our knowledge of user behaviour, problems have to be solved concerning the ways in which that knowledge can or should be applied in the process of system design and development. Two interrelated issues are pertinent. The first concerns the kinds of conceptual and empirical tools which can be developed to assist in the process of designing systems which are "easy to use". The second concerns who should be the primary user of the tools - should it be system designers themselves, the human factors practitioner, or some amicable division of responsibility between the two forms of expertise? Involved in this second issue is the important question of the extent to which system designers should be trained, or otherwise acquire, knowledge of human factor and its techniques.

The possibilities for developing a number of different kinds of design aids or "tools" are currently being explored. These include providing the designer with human factors "guidelines", procedures and checklists, or more sophisticated methods for estimating user performance. Empirically oriented aids include "benchmark" tests for usability and methods for "debugging" the user problems with prototype interfaces. Of these, "guidelines" are the most familiar way for presenting human factors information and are frequently requested by systems designers. The prospects for dialogue design guidelines will be critically evaluated on three criteria - their interpretability, their "truth" and their applicability. Examples of problematic guidelines will be discussed on the basis of these criteria. In spite of the difficulties with the guideline approach, it is obviously important for system designers to have some knowledge of the scope and nature of human factors information and methods. Such knowledge would enable them not only to use some behavioural tools for themselves but also to communicate and co-operate productively with human factors practitioners. Some potential problems and pitfalls of the current state of designer "knowledge" of user behaviour will be illustrated by reference to research on the ways in which designers appear to make their decisions concerning user dialogues and models.

Applying the Products of Research on Interactive Dialogues

Philip Barnard

MRC Applied Psychology Unit
Cambridge

The general claim made by behavioural scientists and human factors practitioners is usually that interface design would benefit from the application of behavioural knowledge and empirical methods. There would seem to be at least four good reasons why computer scientists should at least have a passing acquaintance with the knowledge and techniques: to alter the way in which they arrive at individual design decisions; to enable productive communication with human factors experts or other "user-specialists"; to sharpen their own knowledge of the ultimate end-users of their products; to provide a grounding in behavioural tools and techniques which they may encounter later in their careers.

The first three of these points are well illustrated in the comments of system designers reported by Hammond et al (in press). The comments were obtained using a structured interview technique. This was aimed at exploring the nature of decisions taken by designers concerning usability and their views about human factors in general. Individual design decisions are typically taken against the background of numerous constraints which operate within the development process. Within these constraints, designers have often justified their decisions about interface design by appealing to the constraints of the internal architecture of the system or by appealing to strongly held views about consistency within a product or compatibility with past and future software. It was also not atypical for designers to have possessed little knowledge of human factors. Some of the designers had at least attempted to consult human factors information; but they were by no means always impressed with the results of their efforts. Finally, designers clearly had their own "common sense" theories about users. Such common sense theories, of course, may or may not be accurate. In each of these cases an effective understanding of human factors either directly, or indirectly via human factors experts, could have had a bearing on the outcome of individual design decisions.

Solution Paths and Tools

If human factors information is to be systematically incorporated into the development process, what kind of involvement should be considered; what tools are available; and who should use them? There are at least three solution paths to the incorporation of methods and knowledge. The first reflects the role often played by human factors in the past. Once designed, a completed system is put up for behavioural evaluation. It is too late for many of the key design decisions to be altered. On these grounds human

factors experts have typically argued for an alternative solution path in which their expertise is brought to bear at any early stage in the design process. A third, related, solution path is a kind of apartheid in which there is a division of design responsibility. The computer system specialist should be responsible for designing the internal architecture and functions, while a user-specialist should take care of the interface. On this view, the internal architecture and functions must be functionally separated from the software which controls the actual end-user interface.

The kinds of tools and techniques which can be incorporated into these solution paths include limited simulation and behavioural pre-testing of user problems and performance (eg. see Clark, this seminar). More formalised "benchmark" tests, incorporating a standard set of user tasks with known characteristics for other systems, could also be developed and applied. For cognitive aspects of interface design, calculational methods are under development for predicting user performance at the initial stage of system specification (Card et al, 1983; Card, this seminar). Guidelines and checklists are also under development for interface design as an immediate, pragmatic decision aid (eg. see Smith, 1982). These various approaches all have their strengths and weaknesses, particularly concerning aspects of user cognition where our formalisable knowledge requires a good deal of its own research and devlopment. At present it is a fledgling applied discipline within the behavioural sciences. The actual development and application of such a disipline requires at least three phases. First, there is a need to convince computer system designers of its importance. This in itself requires practical demonstrations of its actual and potential utlity. Second, coherent methods and conceptual tools must be established for application now - whilst better ones are devloped and validated through research and experience. The third phase is "the promised land" where mature behavioural tools and techniques are automatically used as a part of each phase of design and development.

A key role in these phases is played by "our knowledge of human cognitve skills". Without it, our tools may founder. Such knowledge can be embodied in calculational techniques or in guiding "principles". Both must necessarily simplify the complexities of cognition in order to provide practical decision support. Both have deficiencies. In the past, for example, calculational simplification has sometimes had less than satisfactory consequences. "Readability" formulae developed some years ago are often applied to the analysis of document comprehensibility in a relatively mindless way: if its readable, on this measure, its usable (but see Wright, 1980). Such methods may not prove wholly satisfactory for capturing all aspects of the communicative content of user-system dialogues.

The most often requested alternative for embodying human factors knowledge and experience involves guidelines, checklists and standards. These also have deficiences, which are often

acknowledged by the guideline developers themselves. For some aspects of the specification and design of the hardware aspects of interfaces guidelines can have considerable utility (e.g. see Ramsey & Atwood, 1979). For guidelines concerning cognition, a common assumption is that some guidelines are better than none. While relatively straightforward cognitive blunders continue to occur in dialogue design, such guidelines will undoubtedly have some utility. As more experience is gained with user-computer communication, and as more powerful software comes into play, any blunders which could have been prevented by guidelines may decrease. Furthermore, at the time at which they are developed, each new generation of interface techniques are likely to create their own problems which may not be captured by earlier releases of specific guidelines. It therefore seems appropriate to take a serious look at them now with a view to laying the groundwork for more powerful ways of reasoning about the content of future forms of human-computer interfaces and dialogues.

Guidelines

Typically, a guideline is a relatively simple statement coupled with an example, any exceptions to the principle and pointers to other guidelines. If available, source references may be included so that the user of the guidelines can trace its origins to particular systems or research papers. The current "state of the art" is well illustrated by Smith's (1982) recent compilation and discussion of some 375 guidelines for computer based information systems. Of these, 98 concern data entry; 134 concern data display; and 143 concern sequence control. Smith estimates that there may be between fifty and one hundred "archetype" tasks which require analysis and that a comprehensive compilation might run to over one thousand items. The intention is to produce relatively specific guidelines which can be interpreted in the context of a particular design problem. Formal research on which to base the guidelines is not always available and evaluations of their utility in the design process need to be pursued. Smith also cites a survey of people interested in the production and use of guidelines from which it is reported that 73% of the respondents would like to develop more guidelines now on the basis of informed judgement. Only 14% prefered to wait for "data". This is very much in line with the assumption that some guidelines, however inadequate, offer a better prospect for designers than none at all.

Published guidelines are phrased at one of three levels of generality. Very general statements include:

"Know the User" (Hansen, 1971)

"Users learn best by doing"
"The most efficient structures for computers are not necessarily efficient for users"
"Every designer assumes the system is easy to use"
(Hiltz & Turoff, 1978)

Other guidelines attempt to capture some more specific principle about user behaviour or implicit assumptions about the usability of content:

"Nomenclature used is oriented towards, and appropriate for, the application"
"The proper visual ground is present"
"As few demands as possible are made on the user's memory"
(Baecker,1980; describing principles underlying an application)

The more general "words of wisdom" often capture an important point and have undoubted value at times when system design apparently ignores them. They are interpretable only as general exhortations towards a fundamental change in approach or opinion. The guidelines oriented towards statements of particular principles have a similar communicative value: they act to sensitise designers about particular issues which they may not have explicitly thought about before. They may sensitise but, in and of themselves, they do not carry enough information to enable them to be interpretable and applicable by a designer who does not possess specialist knowledge of behaviour. Even with recourse to hard behavioural evidence, the designer may still be unable to resolve a decision about "nomenclature being appropriate to the application" (see discussion of evidence concerning naturalness of text editing terminology in my own earlier presentation in this seminar).

General acknowledgement of these kinds of issues has led to much greater emphasis on the development of very much more specific guidelines:

"If a fixed length word or collection of characters is to be entered via the keyboard, limit the field on the screen by special characters, for example, underscores".
Engel & Granda (1975)

Many of these specific guidelines presuppose a behavioural principle and capture some element of truth more or less accurately for the particular issue addressed in the domain of interface design. These guidelines will be helpful even though they can only be expected to capture the true issue by way of approximation. Other very specific guidelines, particularly those based on judgement or preliminary behavioural evidence, may actually be inaccurate. The inaccuracies may then result in the guideline being used to arrive at design solutions which are actually inappropriate. System designers cannot necessarily be expected to be able to discriminate the good ones from the bad ones. Additionally, bad or poorly motivated guidelines could lead to particular features becoming accepted practice or ossified as a "standard". For example, frequent use of incomprehensible negative sentences has led to the incorporation of the unqualified statement "In textual display, affirmative statements should be used rather than negative statements" (Smith,1982). In fact, there are circumstances where the use of negative forms are

indeed appropriate - particularly to prevent people performing some automatic action (eg. see Wright & Barnard, 1975). Likewise, another guideline, from the same source, states "Lists within text should be ordered by some logical principle; long lists, with more than seven items, should be ordered alphabetically". On this basis a designer must go ahead with a decision to use an alphabetic listing under circumstances where it is known that a superordinate conceptual organisation might considerably enhance search times for users (eg Barnard et al, 1977).

Since guidelines tend to be simple statements, one of the major problems with their applicability is that they do not handle very well the trade-offs between different factors. For example, suppose we wanted to design a command dialogue for an airline booking system. Three potential principles might be relevant. We might consider attempting to make the argument structure compatible with natural language expression such as: BOOK <passenger> (on) <Flight Number> or LIST <Flight no> <date>; Alternatively, we might want to arrange things by placing particular arguments in a constant position to reduce the load on a user's memory. For this principle we might argue that it would be better to always have the flight number placed first. There may also be ordered flight status and passenger information presented on the VDU . In this context, we might want to consider making argument entry compatible with the order of information display fields.

The equation could be further complicated by the size of the command set and by the pattern of usage. For transaction clerks in a major package holiday firm, who would be constantly entering such data, the display compatibility solution might be the crucial factor since a major part of their task might involve scanning. As routine users, the command set size, natural language compatibility, and reducing memory load for the command-argument structure may be less important. Further, the obvious solution of providing argument prompts might slow them down significantly. For occasional users, with a small command set size, prompts or natural language compatibility might be the least error prone. With the kind of memory load imposed by a large set of commands, the constant position principle might act better than the natural language principle to reduce order uncertainty and might therefore be the most profitable course to pursue. Such issues could, of course, be resolved by empirical methods. However, the crucial point is that simple guidelines do little to capture important information for this kind of reasoning process. As such they might have little utility either in design specification or in determining which of a restricted set of options might be subjected to empirical test.

Even the more specific guidelines may thus in the long term not measure up to intricacies of the design decision process. Simple statements are also unlikely to capture the intricacies of human performance as evidenced by experiments on human-computer dialogues and other topics in applied cognitive psychology. Many of these experiments strongly suggest that user performance is

typically task and context dependent. These interdependencies are not easy to capture in simple statements. Many guidelines are stated primarily in terms of the descriptive properties of the style, structure and content of dialogue exchanges, which do not adequately take these factors into account. On these grounds we can expect there to be few simple, effective and enduring guidelines for cognitive aspects of human-computer interaction. Whilst guidelines may have a role to play in the immediate future, in the longer term we should try to establish better means of acquiring, representing and applying our knowledge of user cognition and its consequences.

An "Expert" Solution?

In the longer term, one clear requirement is for conceptual tools which support the process of reasoning in the design context. Such tools should be usable by whoever is designing the user interface, irrespective of whether their specialist skills lie primarily in computer science or in behavioural science. The principles incorporated in the tools should support design reasoning not by a description of external "states of affairs" but by principled reference to the likely attributes of user cognition. This will involve:

(1) A Characterisation of the Users
-Their knowledge and cognitive skills together with their conceptual implications.

(2) A Characterisation of Performance
-The empirical consequences of knowledge and skill in terms of time, errors and strategies.

(3) A Mapping to the specific design context.
- a means of interpreting (1) & (2) in context via a "cognitive", rather than a logical task analysis.

Given what we already know about user cognition, this is not going to be an easy task. It requires a "deep" theoretical understanding of human cognition, The behavioural sciences have not always been good at delivering such an understanding, particularly for applied contexts. The content of the analysis might also prove "user-unfriendly" both to human factors practitioners and designers. One potential route for coping with the complexities of reasoning about human cognition would be to recruit software technology itself and attempt to embody our characterisations in an expert system. To achieve this we would need to specify our characterisations of users and their performance by furnishing principled answers to the following questions:

(1) What system and task relevant knowledge does the user bring to bear in the course of an interaction?
e.g. - of other software
- of the application domain
- of natural communication etc.

(2) How is that knowledge represented?
e.g. - as integrated conceptual "models"
- as knowledge "fragments"
- as simple token sequences which achieve a goal

(3) How do users structure their tasks, goals and strategies for achieving them?

(4) What changes occur in knowledge representations and their accessibility as learning progresses?

(5) What mental architectures and processes mediate performance and how do they constrain it?
e.g. -memory retrieval
-search

(6) In what way do users differ systematically on (1)-(5)?

Such characterisations will of course, be dependent upon particular research or theoretical viewpoints. Nevertheless, it might prove possible to "elicit" sufficient knowledge from cumulative experimentation with constrained areas of user-system interaction and to embody it in a limited but practical decision aid. In the case of the experiments on the use of interactive dialogues which were summarised in my previous presentation, some of the relevant user characterisations which were inferred include:

General

(1) Users' purposes or goals will determine how they mobilise their specific skills and knowledge.

(2) Information encoded in memory will depend on the nature of the task demand for what is to be learned.

(3) Users will strategically recruit information structures in semantic or episodic memory and from the immediate task environment to resolve structural or semantic uncertainty.

.
.
.

Knowledge Representation

(1) For repeated free order selection of dialogue constituents from an external information structure (menu), the inferred user mental representation (internal information structure) is disjunctive.

(2) Disjunctive user representations result in inconsistent user memory representations.

(3) Verb-Object is the probable form of imperative information structures in natural language.

.
.
.

Mappings to the specific design context

(1) A menu is an information structure.

(2) Expressions in documentation or initial instruction are task relevant information structures.

(3) A question (in the user's head) is a task relevant information structure.

(4) Menus mediate recognition access to user representations

(5) Command entry requires active recall.

.
.
.

Characterisation of user performance

(1) Under conditions of order uncertainty and under conditions of recognition access, users' choices of dialogue orders for constituents in a transaction are biassed by information structures from natural language.

(2) Under conditions without order uncertainty and under conditions of recognition access, selection performance is relatively unaffected by transient information structures.

(3) Under conditions of active recall with or without order uncertainty transient information structures may influence performance.

(4) (Conditions), bias from pre-use instruction is relatively transitory.

(5) (Conditions), bias from early acquisition of concepts and procedures through actual use is enduring.

(6) (Conditions), inconsistent episodic memory records cause retrieval and task transfer difficulty.

.
.

Just as with guidelines our initial characterisations may be inadequate by way of inaccuracy or incompleteness. However, in the longer term it may be a more productive way of embodying our knowledge in a decision aid for designers. First, it offers the

promise of supporting reasoned decisions. Indeed, its power to explore potential interactions between multiple factors in the design space may quite quickly outstrip the normal abilities of its own user. Second, individual rules and fragments of knowledge could be cummulatively updated with with research evidence and interpretation. Third, the actual development of such a system may itself be a useful vehicle for communication and collaboration between computer science and the behavioural sciences. As an added bonus, it offers the prospect of theoretical developments within the behavioural sciences that are grounded in "real" behavioural tasks rather than the more artificial tasks of memory and understanding typically employed in laboratory investigations of human cognition.

References:

Baecker, R. (1980) Human-computer interactive systems: A state-of-the-art review. In Kolers, P., Wrolstad, M. & Bouma, H. (Eds.) Processing of Visible Language 2. Plenum Press: New York, 424-443.

Barnard, P., Morton, J., Ottley, E. & Long, J. (1977) Planning menus for display: some effects of their structure and content on user performance. In Displays for Man-Machine Systems. IEE Conference Publication 150 IEE: London; 130-133.

Card, S., Moran, T., & Newell, S. (1983). The Psychology of Human-Computer Interaction. Lawrence Erlbaum: New Jersey.

Engel, S., & Granda, R. (1975) Guidelines for Man/Display Interfaces. IBM Technical Report TR 00.2720.

Hammond, N., Jorgensen, A., MacLean, A., Barnard, P., & Long, J., (In Press) Design practice and interface usability: evidence from interviews with designers. To appear in the Proceedings of "CHI'83 Human factors in Computer Systems"; Boston 1983; ACM.

Hansen, W., (1971). User engineering principles for interactive systems. In AFIPS Conference Proceedings, vol 39, 523-532.

Hiltz, S. & Turoff, M. (1978) The network nation: Human communication via computer. Addison-Wesley: Reading, Mass.

Ramsey, H., & Atwood, M., (1979) Human Factors in computer systems: A review of the literature. Technical Report SAI-79-111-DEN. Science Applications. Inc.

Smith, S. (1982) User system interface design for computer based information systems. Mitre Corporation Technical Report, ESD-TR-82-132.

Wright, P. (1980) Usability: The criterion for designing written information. In Kolers, P., Wrolstad, M., & Bouma, H. (Eds), Processing of Visible Language 2, Plenum Press: New York. 183-205.

Wright, P. & Barnard, P. (1975) Just fill in this form: A review for designers. Applied Ergonomics, vol 6, 213-220.