BEHAVIOUR AND INFORMATION TECHNOLOGY, 1985, VOL. 4, NO. 1, 31–45

Adaptive personalized interfaces—A question of viability

SAUL GREENBERG and IAN H. WITTEN

Man–Machine Systems Laboratory, Department of Computer Science, The University of Calgary, 2500 University Drive NW, Calgary, Canada T2N 1N4

Abstract. It is widely accepted that interfaces between computers and users should differ to accommodate individual, or group, needs. One method of 'personalizing' an interface is to have the system form a limited model of the user and employ it to fashion the dialogue to his needs. Unfortunately, little is known about the effect of adaptation on the man–machine interface. Although obvious advantages accrue from 'personalized' interfaces, there are also obvious disadvantages to presenting users with a changing, adapting and perhaps apparently inconsistent interface. The goal of this work is to determine the viability of an adaptive interface through a human-factor pilot study of a simple, specially designed, interactive computer system.

The system uses menu-driven selection to retrieve entries from a large ordered telephone directory. This simple task has several advantages: it is a realistic application area for interactive computers; plausible adaptive modelling methods exist and have been studied theoretically; and previous work has determined the best way to display the menus to users.

The results of this empirical study support the use of adaptive user modelling. In the (admittedly highly constrained) example system, a computer interface can indeed adapt successfully to every user. Although it does not necessarily generalize to other user interfaces, the result supplies evidence to refute published objections to adaptive user modelling in general.

1. Introduction

User modelling in interactive computer systems is frequently advocated but seldom studied quantitatively. This omission is not due to any serious conceptual barriers but rather to the infancy of the subject. In particular, an important question left unanswered by the literature is whether or not adaptive interfaces can actually help the user. The goal of this work is to determine the viability of an adaptive interface through a human-factor pilot study of a specific interactive computer system. The system was specially selected to favour the use of adaptive methods. If they indeed proved superior in practice, this would refute published objections to adaptive user modelling in general. If not, it would provide strong and quantitative evidence to support the objections.

A *user model* is defined as a set of rules which a computer system follows to determine its reaction to a user. In other words, it is the computer's model of the user. Normally it is formulated by the system designer; in the adaptive interfaces considered here it is further refined by the system itself following procedures set out by the designer. A system is *personalized* when part of this model is unique for each user or group of users. It is *adaptive* when the model is altered during interaction with the user, to reflect a changing view of him.

Innocent (1982) has identified three different strategies for user modelling. The first is when a system designer listens to feedback from the user population and adjusts the system to fit the current need. For example, most office information systems require a system administrator to install and maintain application programs, and to modify aspects of the interface both to improve it and to keep up with office workers' changing needs. Unfortunately, this strategy presupposes a canonical (typical) user, which cannot be an accurate view of a highly heterogeneous community (Rich 1983). Edmonds (1982) suggests that our knowledge about human behaviour is inadequate to portray correctly a canonical user, particularly one whose needs change over time.

The second user-modelling strategy is to let each user modify his working environment himself. An example frequently seen in computer systems is a facility for user-defined abbreviations. Many authors mention the need for a user to be able to tailor a system to his individual abilities, tastes and preferences (see, for example, James 1980, Thimbleby 1980). Unfortunately, there are real disadvantages to explicit personalization. Naive and casual users may find it difficult to learn how to modify their environment—and these are the users who may benefit the most. Furthermore, there is a trade-off between the setup overhead the user is willing to accept and the work he wishes to accomplish (Rich 1983, Greenberg 1984).

Thirdly, there are adaptive modelling strategies in which the system monitors the user's activity and tries to adapt automatically to his model. Although potentially perhaps the most powerful of the three strategies, this method is infrequently used. One of the few examples of systems which attempt adaptive modelling is the *reactive keyboard* (Witten *et al.* 1983 a), which displays a menu of predictions of the user's future keystrokes on the basis of the redundancy which has been exhibited in his input.

The present paper is concerned only with adaptive user modelling. Its structure is as follows. In the next section we review the background to user modelling. This review exposes the fact that it is genuinely not clear whether systems exist which can benefit from adaptive modelling of the user. The following section describes the telephone directory system which was developed for the pilot study. It was considered important to select an application in which user modelling could easily be incorporated, which actually encouraged the use of adaptive methods, and which constituted a realistic application area for interactive computers; yet which was simple enough to study and test thoroughly. The subsequent sections describe the pilot experiment that was run, including the method used, the data collected, statistical analysis of that data and discussion of results. Finally, we draw some conclusions about the experimental results and their likely applicability to other interactive computer interfaces.

2. Background

Many authors imply or state outright that adaptive personalization is desirable. Primary reasons supplied by these authors are summarized in the left-hand column of table 1. Edmonds (1982), Larson (1982) and James (1980) are concerned with variations in user experience, and stress the need for systems that can adjust or be adjusted to a wide range of user ability. Moreover, user needs are constantly evolving, whether on an individual level or through global alterations to the tasks being undertaken. Adaptive interfaces could not only prolong the system's life (Maguire 1982), but may also minimize the frustration experienced by an evolving user of a rigid system. Shneiderman (1980) notes that a satisfying system gives the user the sense of being in control. It is likely that this goal can best be achieved across a heterogeneous user population by adaptive personalization (Thimbleby 1980). Another practical advantage of adaption is that it provides a mechanism to minimize conflicts that arise in designer–user communication, through an automatically updated and corrected user model. Successful interaction between user and designer is traditionally inhibited by

Table 1. Issues in adaptive user modelling.

Pro-adaptation	Anti-adaptation
Variations in user expertise	Dynamics of user–system concurrent modelling
Evolving user needs	User does not have appropriate control
User has appropriate control	Difficulty of implementation
Minimize user–designer conflicts	Inaccuracies of model construction
	Evaluation of benefits

conflicting views, inadequate user models and lack of dialogue prototyping and monitoring tools.

But it is simplistic to assume that adaptive user modelling will necessarily cure a great many ills, for it introduces its own problems (second column of table 1). For example, at the same time as an adaptive system is trying to make a model of the user, the user is trying to model the system. This concurrent effort to create models of each other may lead to instability (Gaines and Shaw 1983). It may undermine a user's confidence in the system, which he expects to be 'consistent and uniform rather than adaptive and changeable' (Innocent 1982). The user's control of the dialogue will be endangered if an automated system prescribes an action which he does not understand (Greenberg 1984). By definition, an automatically adapting interface must have control over some aspect of the dialogue—control which the user may prefer to retain himself.

Yet another disadvantage is the increase in implementation complexity necessary for any system that evolves over time (Eason and Damodaran 1979). In fact, Gaines and Shaw (1983) doubt the ability of an automatically adapting system to gather enough useful information about individuals in time for meaningful action. Models generated by the system may be inaccurate, for automatic adaptation involves guesses about the user (Rich 1983). Finally, it is not clear how one can evaluate an adaptive system empirically (Innocent 1982). There are conflicting opinions of how software quality in general should be measured (Shneiderman 1980). The lack of consensus of useful metrics for evaluation would only be further compounded by the potentially complex adaptive systems.

Are adaptive systems a viable alternative to static ones? Clearly many conflicting issues are involved, and the question can only be answered empirically. No comparative evaluation was reported by any of the authors cited above. Indeed, papers in this area usually present a series of personal views, intuitive feelings, folklore guidelines and implementation ideas, rather than empirical data gathered from experiments on testable hypotheses. The present paper describes a human-factors pilot study designed to prove or disprove the effectiveness of adaptive user modelling in a particular interactive computer system.

3. Telephones and the personalizable directory

In many databases, users tend to retrieve items that have been accessed previously. In other words, they exhibit a repetitive pattern of access. For example, a homemaker consults a small number of recipes in a cookbook repeatedly. A stricken lover favours certain poems in a poetry book over others. A computer user references specific entries in an on-line manual many times over. The actual sets of items retrieved by different users may be disjoint, overlapping or identical; while the frequency of repeated accesses can exhibit high variation across users and across entries.

These characteristics of database access can be modelled as a frequency distribution. The Zipf and closely related Bradford distributions are plausible models of the repetition observed in a variety of real-life usage patterns, such as journal use in a library and frequency of invocation of computer operating system commands (Zipf 1949, Peachey *et al.* 1982). Databases characterized by such distributions seem to encourage adaptive user modelling because there is an opportunity for the query interface to give preference to items that have been retrieved previously. Table 2 suggests desirable properties of personalized retrieval methods, based on a study of user habits for databases with many repeat accesses (Greenberg 1984).

Table 2. Suggestions for personalizing databases which exhibit repeat accesses.

(1)	Ease of access to items should be graded according to their retrieval frequencies
(2)	First time accesses to entries should not be noticeably longer than in a non-personalized system
(3)	Personalization should be done at an individual level
(4)	Personalization schemes should handle a wide variety of usage patterns
(5)	The system should be primed to a sensible default state

3.1. *Characteristics of telephone usage*

Telephone directory usage is a good example of repetitive access to databases. When a telephone number is dialled, it is likely that the caller has used it before. This is no surprise, for each person repeatedly calls a small subset of a very large set of possible telephone numbers. Notice that we are talking here about the pattern of calls made by any one person, and not about the aggregated calling pattern of a population. The frequently used numbers will differ from person to person, but each will exhibit a similarly repetitive access pattern.

In a limited study of personal telephone usage over a period of 2 months, Greenberg (1984) estimated that 50–60 per cent of calls are to numbers that the person has dialled previously. Furthermore, if numbers are ranked by popularity, ranging from very high to very low frequencies, they approximate the Zipf distribution. Figure 1 shows the distribution of calls for each of the three most active subjects studied. The vertical axis gives the number of calls, normalized to one for the most popular number; while the horizontal axis is the rank ordering of popularity. The continuous curve is a true Zipf distribution, normalized in the same way. Of particular interest is the wide spectrum between highly and rarely repeated numbers, shown by the rapid initial decrease of all curves.

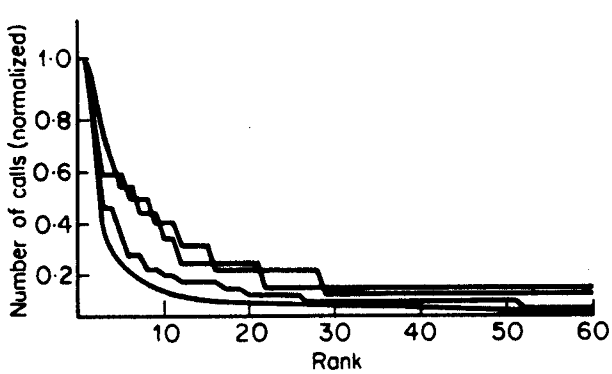

Figure 1. Sample distributions of personal telephone usage, compared with Zipf.

3.2. *Frequency-based menu splitting*

It is quite easy to devise interactive menu-based interfaces which record all calls made by a given user and treat higher-frequency numbers preferentially, at the expense of low-frequency ones. Such a frequency-based menu system provides an attractive way of reducing the average number of selections that a user must make to dial a call (Witten *et al.* 1983 b).

Each menu display comprises a list of ranges of names. The menu divides the name space into regions, one of which is selected by the user (for example, by pointing at it or typing a number). The computer system then calculates a new, second-level menu by splitting the indicated range into subranges. The user makes a further selection, and the procedure repeats. Eventually, some of the menu items will be single names rather than ranges, and when one of these is selected the search terminates.

All the menus, taken together, form a hierarchy. Figure 2 depicts such a hierarchy as a tree, for a small dictionary with 20 name entries. This was obtained by subdividing the name space as equally as possible at each stage, with a menu size of four items. The number following each name shows how may menu pages have to be scanned before that name can be found. The corresponding first-level menu is shown in table 3.

Table 3. First-level menu for figure 2.

(1)	Arbor	–	Eagan
(2)	Farell	–	Kruger
(3)	Kwant	–	Obrien
(4)	Perry	–	Zlotky

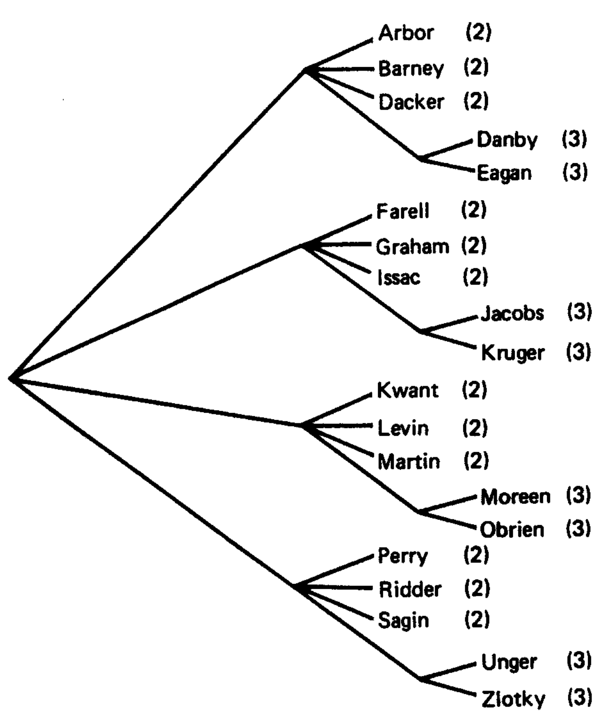

Figure 2. Menu tree generated by uniform subdivision.

Now consider how information about the access frequency of each name may be used to speed up retrieval. The access frequencies define a probability distribution on the set of names. Instead of selecting regions at each stage to cover approximately equal ranges of names, it is possible to divide the probability distribution, reflecting the 'popularity' of the names, into approximately equal portions. During use, the act of selection will alter the distribution by increasing the probability value of names which are selected. Thus the user will be directed more quickly to names which have already been accessed—especially if they have been accessed often and recently—than to those which have not.

Figure 3 shows a menu hierarchy reflecting a particular frequency distribution. Highly popular names, such as Graham and Zlotky, appear immediately on the first-level menu, shown in table 4. Other, less popular, names are accessed on the second-level menu; while the remainder are relegated to the third level.

Given a frequency distribution, it is a surprisingly difficult problem to construct a menu hierarchy which minimizes the average number of selections required to find a name. Exhaustive search over all menu trees, while possible, is infeasible for all but the smallest problems. The problem has been studied and simple splitting algorithms described which achieve good performance in practice (Witten *et al.* 1984).

Table 4. First-level menu for figure 3.

(1) Arbor – Farell
(2) Graham, John
(3) Issac – Unger
(4) Zlotky, Ivan

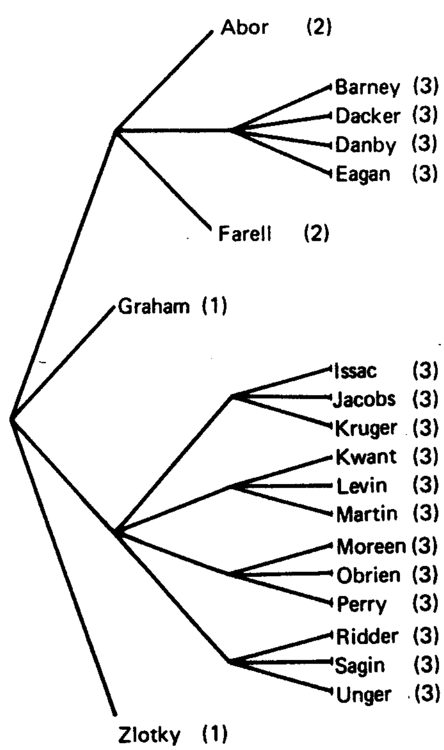

Figure 3. Menu tree reflecting popularity of items.

3.3. *The personalizable directory*

The personalizable directory, as outlined above, provides a simple and definite example of adaptive user modelling. The frequency pattern of calls constitutes a model of the user which can be easily updated as the interaction proceeds (Witten *et al.* 1983 b). The extreme simplicity of the directory task—retrieving items from an ordered list—makes it possible to examine the user modelling problem uncluttered by unnecessary detail. Nevertheless, the system is a realistic one. Menu-based selection is a popular man–machine interface technique for the casual user. Recalculation of the menu at each stage provides significant advantages over the normal method of storing fixed, pre-formatted menus in the information database; for it permits the menu size to be altered to accommodate different display devices. This is particularly important with the increasing use of window-based terminals, in which the size of the display 'window' is not known in advance.

Despite the practicality of the task, however, we do not claim that the personalizable directory system described here is ready for the market. It is not the intent of this study to design a production system. Direct manipulation of a simulated Roladex card file may provide a better interface for both experienced and casual user (Shneiderman 1983). However, the personalizable directory is more than adequate to test the effectiveness of adaptive modelling in a man–machine interface.

4. Comparison of personalized and non-personalized directories

Tools have now been assembled with which to investigate the effectiveness of automatic user modelling through a human-factor experiment. Using the directory retrieval scheme described in the previous section, two systems can be compared which differ only in their user models, simply by maintaining or ignoring the user's access profile. Ignoring the profile gives a static menu tree which will be the same on each retrieval, although different parts of the tree will be explored when seeking different names. Taking the profile into account gives an adaptive system whose behaviour will depend on the user's access history. Both systems are menu based and will appear similar to the user; although of course the contents of the menus will differ.

Another necessary tool is data collected in a study of actual telephone usage. This is required because the probability that a particular call is to a number which has been called before increases over time, sharply at first, and then tailing off to a plateau (at a repetition rate of around 50–60 per cent). In order to perform a realistic experiment, the behaviour of the adaptive system should be studied in its equilibrium state. Unfortunately it takes on the order of a hundred calls to attain this state, and it is beyond our resources to run the experiment for so many trials. Fortunately, the adaptive system can be primed in advance to the equilibrium state, simulating the effect of such a long run. A data segment suitable for use in the experiment has been extracted from the recorded behaviour of a particular, typical user (Greenberg 1984); and this is used both to prime the system and to supply names for subjects to seek.

The exact details of how menus are presented to the user is an important concern in this evaluation experiment. Users will take a long time to select from a menu if the display is difficult to scan rapidly. If the adaptive system is then compared with a static control which has fixed and memorizable paths through the menu tree, the scanning delay will weigh against the adaptive interface. Fortunately, a human-factors investigation of six different menu displays has already been performed (Greenberg and

Witten 1984). It clearly shows which menu format encourages superior human performance in terms of scanning speed and error rate. The recommendations have been followed in the present study. In particular, for each menu item—which in general indicates a *range* of names—only the name which terminates the range is shown. Moreover, the name is not truncated, but given in full.

The earlier study also investigated performance of computer novices and experts in menu search tasks. Novices were found to scan menus more slowly and show greater sensitivity to different display formats than experts. In addition, more variability was present within the novice group than the expert group. The current experiment uses computer experts as subjects in an effort to minimize between-subject variability. The subjects fall into the category of *foreign users*; namely those who have no prior experience of a given system but are familiar with computers in general.

Finally, differences between subject performance in scanning root menus and menus buried deep in the tree are reported by Greenberg and Witten (1984). The results suggest that the traversal of alphabetic menu hierarchies should avoid, as much as possible, descending deep into the tree; for selection efficiency deteriorates with depth. This result favours the personalized directory structure, for it minimized descent into the menu hierarchy.

The comparison of the adaptive personalized directory interface against a static control is now described. It tests for the following null hypothesis:

(1) Personalization of the telephone directory system does not affect selection speed and error rate per individual menu and per trial, where a trial is defined as the complete process of locating the final menu leaf after being supplied with a name.
(2) The order of presentation of the two systems to a subject has no effect on his performance.

The *method*, which constitutes the design and physical makeup of the experiment, is described first. This is followed by the *results*, a statistical evaluation of the generated data. Finally, a *discussion* interprets the results within the context of the personalizable directory.

4.1. *Method*

4.1.1. *Subjects*

The subjects were 26 paid volunteers (university students). All were solicited from senior computer science courses. All were very familiar with the use of interactive computers in general, but none had previous experience with the personalizable directory system—they can be described as 'foreign users'. Some subjects had previous exposure to the menu selection interface: they were randomly mixed with inexperienced subjects.

4.1.2. *Subject use*

The experiment was a two-level (adaptation type) by two-level (order) mixed factorial design (figure 4). Each subject was assigned to both levels of adaptation type and only one level or order, giving a total of 13 subjects per cell.

		Order	
		Adaptive first	Static first
Adaptation type	Adaptive	S1–13	S14–26
	Static	S1–13	S14–26

Figure 4. Mixed factor ANOVA design.

4.1.3. *Apparatus*

A Corvus Concept microcomputer connected to a VAX-11/780 was used to display all material for the experiment on a high-resolution bit-mapped screen. Keystrokes were timed locally on the Corvus so that precise measurements could be made (to within 10 ms). Instructions to subjects were given first verbally and then on line. The database used was the University of Calgary telephone directory, containing 2611 entries, modified so that no entries had common last names.

4.1.4. *Data selection*

Data for the experiment were provided by the first 160 calls made by one subject in actual telephone usage. This profile was randomly mapped onto the directory database. The repetition rate stabilized to approximately 63 per cent after 90 calls. The system was primed with the first 122 calls, and the remaining 38 names called were presented to subjects. Each screen display comprised of nine menu items. This led to an average of four menu selections to retrieve a name in the static system, and an average of 2·7 selections for the adaptive system.

4.1.5. *Design*

Subjects were randomly assigned to one of the two order groups. Each subject of each group was exposed to both menu systems.

4.1.6. *Procedure*

Each subject was exposed to two menu systems, one which used a dynamic adaptation algorithm and the other which did not. Each set presented instructions on how to use that particular system, followed by a practice session of eight trials and a test session of 30 trials. Each trial was composed of five parts:

(1) A name was presented to the subject, who read it and pressed ⟨return⟩. Timing of the trial began.
(2) A menu display appeared below the name and was scanned by the subject. His task was to identify the range or the unique entry which contained the supplied name.
(3) If the name was not a unique entry on the menu, the subject was required to select that menu item which indicated the correct range. The scanning time of the menu was noted. The previous step was then repeated with a new menu display representing a branch down the menu tree.

(4) If the name was a unique entry, the subject was required to select it, successfully ending the trial. Timing of the trial ended.
(5) The system indicated errors to the subject, and asked him to make another selection. Errors and their time of occurrence were marked.

The same sequence of 38 names was presented to all subjects during both static and adaptive sessions. As the menu system used forced choice selection, the sequence of menus was identical across subjects, although they differed between static and adaptive sessions.

4.1.7. *Measures*

The dependent measures are the scanning time per trial, error rate per trial, scanning time per menu and error rate per menu. The independent measures are adaptation type and presentation order. Scanning time per trial is the time between the appearance of the first menu and the successful location of the name in the final menu. Error rate per trial is the percentage of selection errors made in a given trial. Scanning time per menu is the time between the appearance of a menu and the successful location of the range containing the name in that menu. Error rate per menu is the percentage of selection errors made in a given menu. In addition, a questionnaire was given to subjects asking them which scheme they preferred, for comparison with the quantitative result.

4.1.8. *Motivation*

It was desirable to keep the subjects' motivation level constant to avoid uncontrolled variation within a subject. A cash prize was therefore awarded to the subject with the best performance, where performance is a combination of error rate and scanning time.

4.2. *Results*

Scanning speed and error rate were analysed independently through the use of the analysis of variance statistical package (P4V) supplied by BMDP statistical software (Dixon *et al.* 1981). Table 5 gives an ANOVA summary table of all results. All starred F-ratios in the table indicate a statistically significant result at or beyond the 0·01 level. For those unfamiliar with ANOVA concepts and terms, a detailed explanation can be found in Kirk (1968).

4.2.1. *Scanning speed and error rate per trial*

A 2×2 analysis of variance on scanning speed per trial revealed a significant main effect for adaptation type and an interaction between order and adaptation type (table 5).

Post-hoc comparisons among cells of the adaptation type by order interaction matrix, using tests of simple main effects, showed significant differences for adaptation at both levels of order, whereas order was only significant with the static adaptation type. Figure 5 illustrates the magnitude of the adaptation type–order interaction. There is a relatively small effect of order, from 0 to 2 s, contrasted with the large effect of adaptation type, from 5 to 7 s. The nature of the interaction allows the consideration, for all practical purposes, of the main effect of adaptation independent of the

Table 5. ANOVA summary table for all dependent variables.

Dependent variable	Source	Degrees of freedom	Sums of squares	Mean squares	*F*-ratio
Scanning speed per trial	Between				
	Order (O)	1, 24	10·629	10·629	1·81
	Subject within groups		141·055	5·877	
	Within				
	Adaptation type (A)	1, 24	419·809	419·809	452·91*
	A × O	1, 24	14·217	14·217	15·34*
	A × subject within groups		22·246	0·927	
Error rate per trial	Between				
	Order (O)	1, 24	167·545	167·545	0·93
	subject within groups		4324·110	180·171	
	Within				
	Adaptation type (A)	1, 24	2777·220	2777·220	35·62*
	A × O	1, 24	85·504	85·504	1·10
	A × subject within groups		1871·214	77·967	
Scanning speed per menu	Between				
	Order (O)	1, 24	0·616	0·616	1·20
	subject within groups		12·311	0·513	
	Within				
	Adaptation type (A)	1, 24	0·389	0·389	7·26*
	A × O	1, 24	0·942	0·942	17·58*
	A × subject within groups		1·287	0·536	
Error rate per menu	Between				
	Order (O)	1, 24	11·962	11·962	0·77
	subject within groups		373·356	15·556	
	Within				
	Adaptation type (A)	1, 24	76·182	76·182	12·51*
	A × O	1, 24	4·373	4·373	0·72
	A × subject within groups		146·184	6·091	

*$p < 0{\cdot}01$.

interaction. The cell means of significant main effects on scanning speed are summarized in table 6.

Similarly, a 2 × 2 analysis of variance performed on error rate per trial revealed a significant main effect for adaptation type (table 5). The cell means of significant main effects on error rate are summarized in table 6.

4.2.2. *Scanning speed and error rate per menu*

A 2 × 2 analysis of variance on scanning speed revealed a significant main effect for adaptation type and an interaction between order and adaptation type (table 5).

Post-hoc comparisons among cells of the adaptation type by order interaction matrix, using tests of simple main effects, showed significant differences for adaptation at the static first level of order, whereas order was only significant with the static adaptation type. Figure 6 illustrates the magnitude of the adaptation-type–order interaction. The nature of the interaction precludes analysis of any significant main effects.

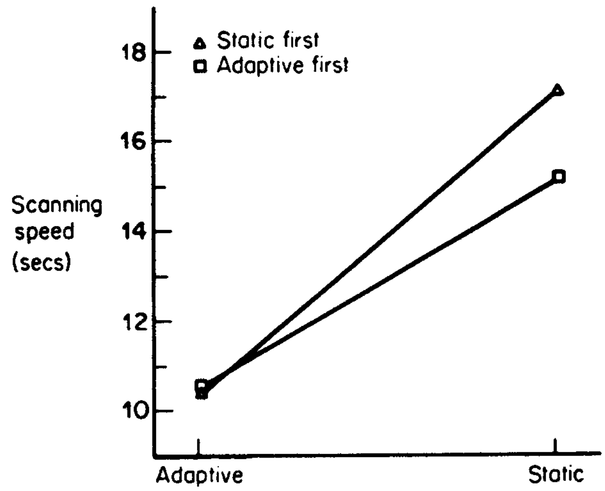

Figure 5. Order–adaptation interaction (trials): scanning speed versus adaptation type.

Similarly, a 2 × 2 analysis of variance performed on error rate per menu revealed a significant main effect for adaptation type (table 5). The cell means of significant main effects on error rate are summarized in table 6.

After completing the task, subjects were asked which system they preferred. Eighteen subjects reported a strong preference for the adaptive system, three were undecided and five weakly preferred the static system.

Table 6. Cell means of significant main effects and their levels.

Factor	Level	Type	Scanning speed (seconds)	Error rate (per cent)
Adaptation	Adaptive	Trials	10·37	10·26
Adaptation	Static	Trials	16·06	24·67
Overall		Trials	13·22	17·56
Adaptation	Adaptive	Menus	—	3·79
Adaptation	Static	Menus	—	6·31
Overall		Menus	3·93	5·01

4.3. *Discussion*

This study compares two systems which differ in their ability to adapt to user input. Two levels of dependent variable are examined: a task (trial) level which is the successful location of a unique name through a chain of menus, and a menu level which is the interim search through any single menu. Evidence supporting rejection of the stated null hypothesis is as follows.

(1) The adaptive directory has a significantly faster trial completion time than the static directory (table 6).
(2) The adaptive directory has significantly fewer errors per trial than the static directory (table 6).
(3) Trial completion time is reduced in the static directory when the subject has prior experience with the adaptive directory. However, the magnitude of improvement is much less than that obtained by use of the adaptive system (figure 5).

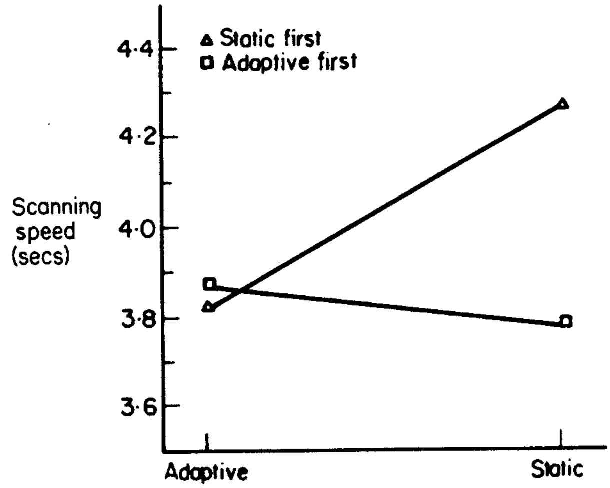

Figure 6. Order–adaptation interaction (menus):scanning speed versus adaptation type.

(4) Menus generated in the adaptive directory are initially faster to scan than static menus. However, no difference exists when the subject had previously used the alternate system (figure 6).
(5) Subjects made significantly fewer errors using menus in the adaptive scheme than using those of the static one (table 6).
(6) Most subjects preferred the adaptive directory scheme. Their comments indicate that this is due to the apparently shorter search paths for repetitive names, the lack of any need to memorize and the frequent appearance of the given name as a range delimiter in some of the high-level adaptive menus, even though further descent into the tree was still necessary.

The results of this experiment indicate that the personalizable directory system is superior to the non-personalizable one, at both the trial and menu levels. Table 7 summarizes the degree of improvement of the personalized system over the static one. The task of locating a given person is shortened by almost 6 s (a 35 per cent improvement). The error rate decreases from 25 to 10 per cent (a 60 per cent improvement). Part of this improvement is due to the subject having to scan fewer menus to retrieve a name in the adaptive system. However, the actual number of menus scanned decreases from 4·0 to 2·7—only a 32 per cent reduction—and this indicates that additional factors contribute to the superiority of the personalized directory.

Table 7. Degree of improvement (static to adaptive).

Task	Percentage reduction (static to adaptive)
Scanning time per trial	35
Errors per trial	60
Scanning time per menu	0
Errors per menu	40
Menus searched per trial	32

At the menu level, fewer errors are made per menu in the adaptive system than in the static one, with a reduction from 6·8 to 3·8 per cent per menu (a 40 per cent improvement). There are two possible reasons for this reduction. First, a subject's attempt to memorize pathnames in a static system may increase errors. Secondly, as found previously, performance degrades with depth in the menu hierarchy (Greenberg and Witten 1984). This condition is minimized in the personalized system.

Subjective preference was heavily weighted toward the adaptive scheme, even though subjects were not told that with it they traversed fewer individual menus. Thus both quantitative and qualitative results favour the personalizable directory system over the static one.

The university directory database used in this experiment was much smaller than city directories, which have hundreds of thousands of entries. Personalization provides an even greater performance advantage, in terms of number of menus scanned, with larger directories. With a Zipf distribution of names retrieved, this approaches a 50 per cent reduction in the average number of menus scanned for a large directory (Witten *et al.* 1984), compared with 32 per cent in the tested system. Thus the performance improvement estimated from the experiment is probably conservative when compared with potential real-life applications.

One counterpoint should be mentioned. The subjects ran data simulating about 10 days of telephone usage. A higher volume of data representing a longer time period would presumably have facilitated memorization of the relatively few highly popular entries, thereby reducing times in the static system. In this case, the benefits of personalization would be confined to moderately accessed phone numbers which are not memorized. One can envision a production system inviting its user to select a menu showing the nine most popular entries, deferring to the personalized directory for all others.

5. Conclusions

This work supports, through empirical study of a particular system, the viability of adaptive user modelling in interactive computer systems. Many of the arguments against personalization, as summarized in table 1, have been contested. An adaptive system has been successfully implemented and found superior to a well-engineered non-adaptive version. It has been shown that concurrent modelling in a man–machine system need not negate the benefits of the interface. Inaccuracies in model construction do not necessarily incapacitate the interface. Repetitions within the dialogue can reinforce important parts of the model and reduce the negative impact of rare or erroneous interactions. Finally, it has been shown indirectly that the user does not necessarily feel out of control with an adaptive interface, even though he perceives the system as changing.

One should take care not to overgeneralize from the result. It should be construed as an existence proof rather than as a general recommendation. The personalized directory application was specially chosen as a likely candidate for adaptive user modelling. The arguments in table 1 against adaptive user interfaces have not been invalidated; only their universal applicability has been questioned. There certainly exist applications which are ill-suited to the adaptive approach, and for those that are suited to it a poorly designed adaptive component will surely degrade rather than enhance the interface. Nevertheless, existence proofs have their uses: in countering dogma and folklore; and in encouraging others to investigate new approaches.

There is no need to limit adaptive user modelling to menu interfaces or databases exhibiting repetitive accesses. It has been shown to be possible for the computer to adapt to every user—albeit in a highly constrained manner. But little is known about the nature of human reaction to more general adaptive interfaces; further empirical studies are needed. The authors believe that such studies will provide data to support the design of comprehensive adaptive dialogues of many types for users of interactive systems.

Acknowledgments

This work was supported by the National Sciences and Engineering Research Council of Canada.

References

DIXON, W. J., BROWN, M. B., ENGELMAN, L., FRANE, J. W., HILL, M. A., JENNRICH, R. I., and TOPOREK, J. D., 1981, *BMDP Statistical Software* (Berkeley: University of California Press).

EASON, K. D., and DAMODARAN, L., 1979, Design procedures for user involvement and user support. *Infotech—Man Computer Commynications*, London.

EDMONDS, E., 1982, The man-computer interface: a note on concepts and design. *International Journal of Man–Machine Studies*, **16**, 231–236.

GAINES, B., and SHAW, M. L., 1983, Dialog engineering. In *Designing for Human-Computer Communication*, edited by M. E. Sime and M. J. Coombs (London: Academic Press), pp. 23–53.

GREENBERG, S., 1984, User modeling in interactive computer systems. MSc. thesis, Department of Computer Science, University of Calgary, Calgary, Alberta.

GREENBERG, S., and WITTEN, I. H., 1984, Comparison of menu displays for ordered lists. *Proceedings of the Canadian Information Processing Society National Conference*, Calgary, Alberta, May.

INNOCENT, P. R., 1982, Towards self-adaptive interface systems. *International Journal of Man–Machine Studies*, **16**, 287–299.

JAMES, E. B., 1980, The user interface. *The Computer Journal*, **23**,

KIRK, R., 1968, *Experimental Design: Procedures for the Behavioural Sciences* (Belmont, Calif.: Wadsworth).

LARSON, J. A., 1982, *End User Facilities in the 1980's*. (Los Almitos, Calif.: I.E.E.E. Computer Society Press).

MAGUIRE, M., 1982, An evaluation of published recommendations on the design of man–computer dialogues. *International Journal of Man–Machine Studies*, **16**, 237–261.

PEACHEY, J. B., BUNT, R. B., and COLBOURN, C. J., 1982, Bradford–Zipf phenomena in computer sytems. *Proceedings of the Canadian Information Processing Society National Conference*, Saskatoon, Saskatchewan, May, pp. 155–161.

RICH, E., 1983, Users are individuals: individualizing user models. *International Journal of Man–Machine Studies*, **18**, 199–214.

SHNEIDERMAN, B., 1980, *Software Psychology*. (Cambridge, Massachusetts: Winthrop).

SHNEIDERMAN, B., 1983, Direct manipulation: a step beyond programming languages. *Computer*, **16**, 57–69.

THIMBLEBY, H., 1980, Dialogue determination. *International Journal of Man–Machine Studies*, **13**, 295–304.

WITTEN, I. H., CLEARY, J. G., and DARRAGH, J. J., 1983 a, The reactive keyboard: a new technology for text entry. *Proceedings of the Canadian Information Processing Society National Conference*, Ottawa, Ontario, May, pp. 151–156.

WITTEN, I. H., CLEARY, J., and GREENBERG, S., 1984, On frequency-based menu-splitting algorithms. *International Journal of Man–Machine Studies*, **21**, 135–148.

WITTEN, I. H., GREENBERG, S., and CLEARY, J., 1983 b, Personalizable directories: a case study in automatic user modelling. *Proceedings of Graphics Interface 83*, Edmonton, Alberta, May, pp. 183–190.

ZIPF, G. K., 1949, *Human Behaviour and the Principle of Least Effort* (Ontario: Addison-Wesley).