

Complexity of user interfaces: Can it be reduced by a mode key?

JOCHEN MÜSSELER* , CRISTINA MEINECKE** , and JOHANNES DÖBLER*

* Max Planck Institute for Psychological Research, Leopoldstraße 24, D-80802 München, Federal Republic of Germany; e-mail: muesseler@mpipf-muenchen.mpg.de

** Institute of Psychology, University of Munich, Leopoldstraße 13, D-80802 München, Federal Republic of Germany; e-mail: meinecke@mip.paed.uni-muenchen.de

Behaviour & Information Technology

Publisher: Taylor & Francis Group

Issue: Volume 15, Number 5 (1996)

Pages: 291 – 300

URL: <http://www.informaworld.com/smp/content~content=a713803120&db=all>

DOI: 10.1080/014492996120076

Abstract. Control panels of computer and other modern instruments are often equipped with so-called mode keys, the pressing of which changes the function of other control elements. Thus user keys have different functions depending on the current mode of the instrument. The question is, however, whether it is more userfriendly to have a panel with parallel arranged keys for each function (thus with almost direct possibility to intervene), or to have serial equipment with only a few user keys, where the different functions are only available if the user calls up the relevant machine mode (e.g. displayed on a monitor that operates with different switchable screens). In this case there exist only serial access possibilities. Two experiments compared performances with three types of user interfaces with and without mode keys on the basis of selection times and errors. Although mode keys apparently reduce the complexity of the user interface, our results show that they lead to slower and more often incorrect usage. However, the amount of practice was a moderator variable. As a consequence, for occasional users it is worth considering a less complex interface, that is, with mode keys, but for expert users an interface where each function has its own key should be preferred.

1. Introduction

With rapid spreading of modern microelectronics and the trend toward compact instruments, so-called mode or menu keys on user interfaces are increasingly employed. Mode keys are actually 'empty' keys, their function being to switch the other control keys into another mode; as a consequence one control element can fulfil different functions.

Figure 1a presents a fictitious user interface of an automobile cockpit. The upper key is the mode key; the five keys beneath it are termed icon keys. By pressing the mode key, one changes the functions of the icon keys. Here, the mode key can be used to activate five different states of the interface (service, ventilation, windscreen wipers, lights

and doors/windows). So one can select 25 different functions with only six keys.

Mode keys are frequently found on computer interfaces and on instruments that perform a great number of functions but do not have enough space to assign a specific user element to each function (e.g. remote controls, pocket calculators, car radios, telefax machines, but also industrial photocopiers, machine tools, or control rooms). Thus, with mode keys, one can realize several functions within a small space and with only a few user control elements. Mode keys therefore permit economic production, too.

However, the question arises how a mode key impacts on the user friendliness of an interface. Several factors could have a (positive or negative) impact:

- Number of control elements visible. As already mentioned above, the number of user elements is considerably reduced when a mode key is integrated (in our example, from 25 to 6). At first sight this reduces the complexity of the interface. And, indeed, in stimulus displays where the target element does not differ by a unique feature (such as colour or orientation), an increase in search time is found with increasing number of items (see, e.g. Treisman and Gelade 1980). Likewise, studies on the design of computer interfaces confirm that there is a dependence of selection times and errors on the number of menu items (Kiger 1984, Müsseler 1994, Landauer and Nachbar 1985, Arend *et al.* 1987, Shinar and Stem 1987).
- Access to spatial information. On interfaces with a mode key, the content of the icon key changes with the current mode of the interface. As a consequence, several functions are located at *one* position in the key panel. A clear and, above all, unique assignment between location and key function is not provided and

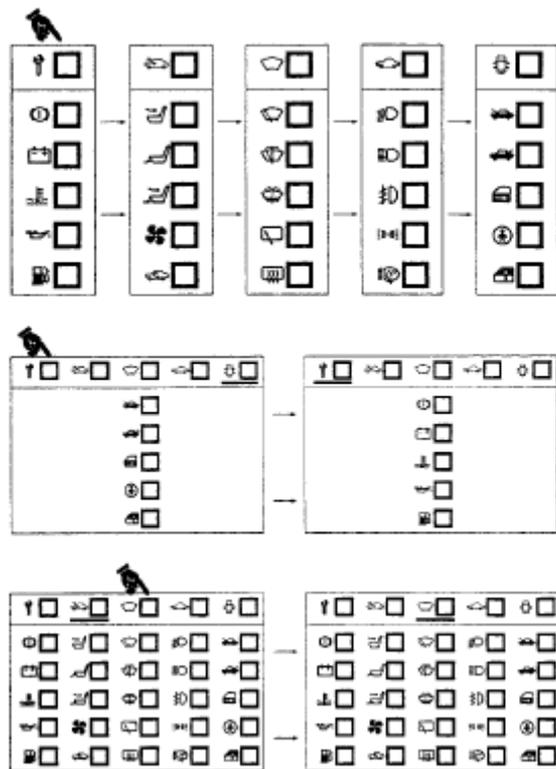


Figure 1a (top part). A car cockpit as an example of a fictitious user interface with a mode key. With the upper mode key five successive categories can be selected, each with five different functions and their corresponding lower keys. The figure displays the five different modes of the user interface. Because of the key arrangement, this interface is labelled with the letter *i*. Figure 1b (middle part). In this interface each category is represented by a separate 'mode' key. The content of the subordinate five icon keys changes through the selection of the specific mode key. The selected mode key is marked by a red horizontal line. This interface is labelled *T*. Figure 1c (bottom part). In this interface each category is represented by an individual 'mode' key and each function by its own icon key. The selected mode key is marked by a red horizontal line. This interface is labelled *Q*.

this could be critical. It has been shown, however, that a spatial alignment increases the accessibility of concepts in an internal cognitive model (c.f. the discussion within the 'mental models' of Johnson-Laird 1983, the 'cognitive maps' of Tolman 1948, McNamara 1986, or the spatial aspects in an assumed working memory, the 'visuo-spatial sketch pad' (VSSP) of Baddeley 1986).

- Access to categorical information. On interfaces with one key for each function, it is possible to group the keys according to their inherent categories. For interfaces with a larger number of items, Paap and

Roske-Hofstrand (1988) recommend that the items be spatially grouped into easily distinguishable categories. Studies by McDonald *et al.* (1983) have shown that categories well-known to the user lead to shorter search times. This possibility of categorical grouping is only successively available on an interface with one mode key, such as illustrated in figure 1a. Here, grouping is attained only by the fact that different categories can be called up successively by activating the mode key. A well-balanced proportion is indicated between the available number of items on the interface (breadth) and the number of invisible items (depth; c.f. Landauer and Nachbar 1985, Tullis 1985).

The purpose of the present study is to obtain more behavioural information on using interfaces with mode keys. The question is how different mode-key realizations influence the speed and accuracy of handling. Hypotheses can be proposed in both directions. On the one hand, the smaller number of keys could facilitate handling. On the other hand, the deficits in spatial and categorical assignment could make it more difficult. These questions are also discussed in the context of supervisory control tasks like they exist in control rooms (for overviews see Becker 1990, Sheridan 1988). One problem here is how to design a control system, that is, whether it is better to have a parallel arranged user panel with almost direct access possibilities, or to have serial equipment with only a few user keys, where the different functions are only available when the user calls up the adequate machine mode (e.g. displayed on monitor operating with different switchable screens). Up to now there is no definite answer to this question.

The following experiments are not primarily concerned with the subjective evaluation of different mode-key interfaces, but rather with which interface causes fewer difficulties (see Eason 1984). As indicators, we use the time the subject required to find a particular function and the numbers of errors made. The first experiment focuses on the behaviour of practiced users; the second experiment then simulates occasional users. Thereafter we will relate the results to the elementary operations postulated, for example, within Fitts' law (Fitts 1954, Fitts and Peterson 1964; for overviews see Rosenbaum and Krist 1996, Müsseler *et al.* 1996) or within the keystroke model (Card *et al.* 1980; for an overview see Reitman-Olson and Olson 1990).

2. Experiment 1

Three different user interfaces were realized, each with the same 25 functions which were assigned to five categories. Thus, the interfaces differed in number and spatial arrangements of the keys, but not in the number of available functions. With reference to the spatial arrangement

of the keys, the interfaces were labelled as *i*, *T*, and *Q* interface.

- The *i* interface. This interface, already sketched, is the typical mode key interface. Figure 1a displays the five different modes that can be activated one after the other by pressing the mode key. Hence, it can be necessary to press the mode key several times in order to get the desired function. For example, if the user is in the mode 'service', he/she has to press the mode key four times to activate the function 'bonnet hook'. In experiment 1, the sequence of interface modes is constant; that is, if one overshoots, a correction always takes four more mode key presses. In the following, this interface is labelled the '*i*' interface, because the arrangement of its columns resembles the letter '*i*'.
- The *T* interface. In the interface displayed in figure 1b, the corresponding surfaces can be directly selected through five permanently visible mode keys. Therefore, the interface comprehends more visible information of the available categories and each of the five modes can be activated by their own key. In contrast, the icon keys of the *i* environment remain: they change their content as a function of the activated mode key. In contrast to the *i* interface the desired function can be selected after only one press of the relevant mode key. Likewise, an immediate correction is possible, because there is no rigid sequence in the surface presentations. This interface is labelled *T* interface, its line and column arrangement resembling the letter '*T*'.
- The *Q* interface. In the *Q* interface displayed in figure 1c, all 25 functions are represented by their own key. The function of the five 'mode keys' is restricted to 'unlocking' the subordinate levels. The assignment of the individual functions to the keys remains constant and is permanently visible. This design is found on interfaces where an incorrect selection has to be ruled out by a two-stage selection (e.g. interfaces with high security demands). Although such interface is rare in everyday use, we included it in the present study to ensure comparability of the motoric demands of all three interfaces. Because of the quadratic arrangement of the keys, this interface is labelled *Q*.

This experiment was designed to investigate how the different mode-key realizations influence the speed and accuracy of handling. An analysis of the three interfaces results in the following picture.

The *i* interface has the lowest number of user elements; this could facilitate the selection of the correct key. However, if a selection is accompanied by a categorical decision, this low number of user elements, and especially the lack of five categorical mode keys, could be a disadvantage compared with the other two interfaces. An exploitation of the hidden categorical information is only possible when

subjects have a mental representation of the sequence of mode changes elicited by pressing the mode key. The *Q* interface is the exact opposite of the *i* interface. On the one hand, it appears to be much more complex and confusing considering the larger number of user elements. On the other hand, it provides the possibility of directly selecting the desired category, and it is also possible to exploit the spatial position of the mode and icon keys. This could reduce selection times and errors. Against this background, the *T* interface is a compromise. It has a relatively small number of user elements that are still clearly laid out in one row and one column. Further, this interface gives the opportunity to make a direct categorical selection.

2.1. Method

2.1.1. *Apparatus and icons*: The experiment was carried out on a PC with a touchscreen monitor (14" NEC Multisync 3D with TekTronix Touchscreen). The resolution of the monitor was set at 640 × 480 with 16 colours. The spatial touch sensitivity corresponded approximately to the screen resolution. Each touch and the corresponding selection time was registered in a logfile.

The user interface represented a fictitious user panel for a car (see figures 1a to 1c).¹ The icons were displayed in black on white (40 × 40 pixels) and were created with reference to the German industrial norm DIN 30 600 (DIN 1988). An accompanying key (also 40 × 40 pixels) was placed to the right of each icon. Every touch on such a key was indicated by a change in colour and a brief auditory feedback ('click').

Three key arrangements were operationalized as shown in figures 1a to 1c. The mode keys were displayed in the upper section of the user panel and were separated by a horizontal line. In interfaces *Q* and *T*, each preselected mode key was distinguished by a red bar below the key (see, e.g. figure 1b: doors/windows or service). The icon keys were located in the panel below them (5 icon keys in the *T*-type and *i*-type interface; 25 in the *Q*-type interface).

The lower field of the monitor displayed the start key as well as an image of the icon that the subject had to activate. This lower field was coloured turquoise in order to separate it from the actual user interface.

2.1.2. *Procedure*: Before the experimental session, subjects were given approximately five minutes to study the meaning of the icons and their corresponding categories on a printout. Then instruction required them to press the start key in the lower field of an otherwise empty screen. The icon the subjects had to select and to activate was displayed, remaining visible until the end of the trial. They were asked to release the start key when they were sure about the meaning of the icon and its category. On release of the start key, the interface was displayed in a randomly selected

starting mode. After touching the correct icon key, feedback was a short sequence of two tones. If selection was incorrect, there was no sequence of tones, and the subjects had to continue until they had found the current path to the target icon. The next trial began when the start key reappeared.

Subjects were instructed to select each icon with as few errors as possible and to take the 'shortest' path. If, for example, the correct mode key was already activated, when the interface appeared, they should touch the icon immediately. In all other cases, it was necessary to touch the mode key at least once.

All subjects were confronted with the three interface types in blocks, and prior to each block subjects practised the specific interface type in 10 randomly-selected trials, which were not included in the analysis. Subjects were tested in three sessions on three days, each session lasting about 45 minutes.

2.1.3. *Design:* Each interface type was presented in blocks to all subjects. The sequence of the block presentations was randomized across sessions and subject. Each of the 25 randomly-presented icons had to be selected three times per type of interface. As three sessions were carried out, this produced a $3(\text{sessions}) \times 3(i, T, Q)$ design with complete repeated measures. Each cell in the design was occupied by 75 observations per subject.

2.1.4. *Data analysis:* The *total time* defined as the sum of all times between touching the start key and touching the correct icon were calculated. This analysis also included those trials where the correct icons were not selected directly but after 'detours'. Alongside the number of errors and the total time, three sequential subtimes were analyzed. The *start time* was defined as the time spend touching the start key and reflects the first inspection of the target icon of the actual trial. The *mode time* was defined as the time between releasing the start key and touching the mode key (precisely, the onset of pressing the mode key). Finally, the *icon time* was the onset interval between touching the mode key and touching the icon key. Subtimes with extreme durations (≥ 4000 ms, 7.6% of the complete data set) were dropped from analysis.

2.1.5. *Subjects:* Five female and seven male college students attending the University of Munich were paid for their participation. Their mean age was 27.1 years.

2.2. Results

2.2.1. *Total time:* A first, rough analysis computed the total

time for each interface that the subjects needed to solve their tasks (sum of all times between touching the start key and touching the correct icon). Thus, this analysis also included those trials where the correct icons were not selected directly but after 'detours'. Although the Q interface seemed to be more complex because of the large number of keys, performance was quickest with this interface: it took an average of 2,266 ms to touch each icon correctly. The T interface took 2,668 ms, and the i interface took the most time with 3,596 ms. Analysis of variance with subsequent Scheffé tests for individual comparisons revealed significant differences between these conditions, $F(2, 22) = 122.06$, $p < 0.001^2$. The long times on the i interface were mostly due to the fact that even a correct use of the mode key in most trials required several touches (up to four) to reach the desired icon; in contrast, only two touches were necessary on the other interfaces.

The more detailed time and error analysis inspected selection times and errors on 'comparable' surface modes and thus also with comparable demands: these were the surface modes where the target icon was activated with exactly *one or two* touches. One touch was enough to solve the task when with the appearance of the user screen the correct category (i.e. mode key) was already activated and the target icon could thus be selected directly (task length 1). A task could be solved with two touches when the mode key had to be touched only once before activating the target icon (task length 2). In order to get interpretable time profiles the following analysis of selection times included only correct trials. A task was classified as incorrectly solved when subjects touched more keys than absolutely necessary.

Task length 1. An ANOVA with repeated measures was computed for the start times and the icon times. Statistical effects were found for the icon times but not for the start times. The icon times not only revealed decreases over session (session 1: 1,532 ms; session 2: 1,229 ms; session 3: 1,211 ms), $F(2, 22) = 18.05$, $p < 0.001$, but also differences between interfaces, $F(2, 22) = 6.14$, $p = 0.018$. As there is no interaction between the two factors, we focused on the differences between the interfaces (see figure 2). Icon times were shortest in i (with a mean of 1,175 ms), whereas they did not differ between T and Q according to a Scheffé test (1,435 ms vs 1,362 ms, $\text{Diff}_{\text{crit}, 5\%} = 201$ ms).

A similar trend was observed with respect to the errors. Their frequencies were calculated for each subject and for each interface type and then transformed to error rates (p). In task length 1, possible sources of errors were unnecessarily touching the mode key or selecting an incorrect icon. There was no reduction in the number of errors across sections, but the interfaces differed from each other significantly, $F(2, 22) = 17.15$, $p < 0.001$. The mean error rate was 0.08 on i , 0.29 on T , and 0.32 on Q . Most errors owed to touching the already activated mode key (75% of incorrect trials on i , 83.6% on T , 87.8% on Q). On i ,

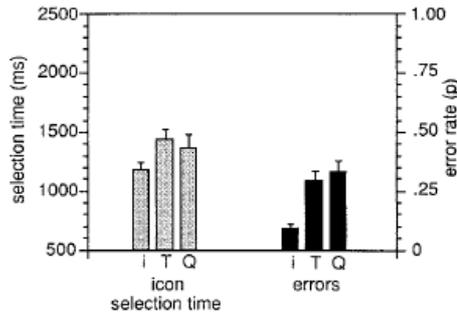


Figure 2. Icon selection times (left ordinate) and error rates (right ordinate) in task length 1. Means and standard errors are averaged across the three sessions (experiment 1).

unnecessarily touching the mode key led to a deactivation of the preselected correct category.

Task length 2. As for task length 1, an ANOVA with the factors session and interface type was computed, but we also included the mode time alongside the start time and icon time. A clear reduction of all times across sessions reflect a learning effect in handling the interfaces (for the start times: $F(2, 22) = 10.14, p = 0.006$; for the mode times: $F(2, 22) = 7.82, p = 0.014$; and for the icon times: $F(2, 22) = 19.71, p < 0.001$). Again, the differences between the types *i*, *T* and *Q* were of greater interest to us in the present context; while the start times revealed no differences ($p > 0.10$), the *i* interface (845 ms) already proved to be inferior to *T* (788 ms) and *Q* (714 ms) in the mode times, $F(2, 22) = 7.98, p = 0.004$, $\text{Diff}_{\text{crit}, 5\%} = 86$ ms (see figure 3). This effect was even more drastic in the icon times: *i* had the highest selection times (1,200 ms), followed by *T* (968 ms) and *Q* (693 ms), $F(2, 22) = 63.31, p < 0.001$, $\text{Diff}_{\text{crit}, 5\%} = 118$ ms.

In the errors, only the main effect of interface was significant, $F(2, 22) = 8.06, p = 0.011$. The mean error rate was 0.14 in *i*, 0.08 in *T*, and 0.05 in *Q*. The characteristic error in *i* was overshooting, that is, unnecessary further touching of the mode key, this led to a deactivation of the already preselected correct category (80% of the errors). In conditions *T* and *Q*, there was an almost equal distribution across several possibilities of errors (e.g. incorrect selection of the mode or icon key; in *Q*, touching the icon key before activating the correct mode key, etc.).

2.3. Discussion

The results of experiment 1 can be summarized as follows: subjects generally required significantly more time to solve tasks in the *i*-type interface than in the *T* or *Q* interfaces. This is evident both in total times and in times of task length 2 in which the mode key has to be activated in

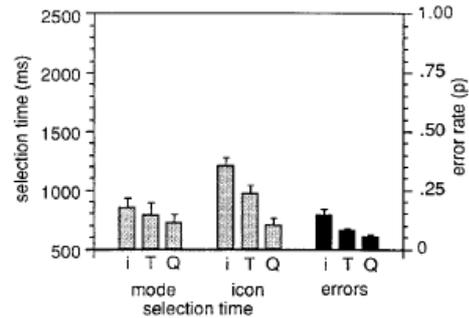


Figure 3. Mode and icon selection times (left ordinate) and error rates (right ordinate) in task length 2 (experiment 1).

order to solve the task. Only if a direct selection of the icon is possible (task length 1), this finding is inverted and processing the interface *i* takes the least time. This pattern of results as well as subjects' verbal reports provide possible indications on specific behaviours in icon search:

- *i interface.* It is characteristic of the *i* interface condition that the starting surface differs from trial to trial in order to prevent subjects from planning the whole motor sequence in advance. Interfaces of this kind (e.g. without an automatic reset to a well-defined starting mode) exist in real life situations, too. So subjects' strategy could be to search through the five preset icons, and – on need – browse through the mode keys without having to consider the categorical membership of the icons. One indication of this was the finding that the shortest reaction times and errors occur for task length 1. However, in task length 2 (icon not visible on the surface) subjects had to note that the target icon was not on the surface, and then had to touch the mode key to change to the next surface. They then reoriented themselves and checked through the five new icons again. Compared with *T* or *Q*, this strategy leads to an increase in the mode and icon times. In addition rhythmic and hasty touching of the mode key leads to overshooting and thus to a high error rate.
- *T interface.* To perform a correct and fast reaction on the *T* interface, the appropriate category of the icon had to be preselected. Because of the constant spatial arrangement of the mode and icon keys, this could, to some extent, be planned in advance. However, in task length 1, this made subjects tend to select the mode key even though the correct category was already preselected. This led to the disadvantage in selection times (compared to the *i* interface), and also in the errors which reflected unnecessary key presses of the preselected category.

- *Q interface*. On the *Q* interface, the icon could be located in the 5×5 matrix as soon as the interface appeared. As, usually, the correct category was not preset, the relevant mode key still had to be touched beforehand. In task length 1, this detour led to a disadvantage compared to *i*, because subjects had to register that it was already activated. This produced an increase in selection times, and – as with *T* – mostly the error of touching the preselected mode key again. In task length 2, *Q* was superior to *T*, because the target specification of the subsequent icon key could be prepared while touching the mode key. Selection times were then correspondingly shorter.

In summary, the different user interfaces and possibility of a direct spatial and/or categorical choice have different effects depending on the interface and the task. If no intermediate categorical step has to be selected and the appropriate icon key has to be touched directly (task length 1), a low number of user keys has a positive effect (*i* interface is better than *T* or *Q*). If, in contrast, it is necessary to pass different surfaces, an interface is helpful that allows this direct choice (as in interfaces *T* and *Q*).

The large number of keys on the *Q* interface does not impair performance. If another mode has to be selected, subjects benefit from the fact that at this point in time, the next icon to be selected is visible in the icon-key panel. Presumably these icons in the (relatively complex) 5×5 matrix are easy to find because each key has its own position and this has been learned. The situation established in experiment 1 therefore probably tests the behaviour of skilled users. This raises the question of whether unpracticed users would show a similar behaviour when working on the interfaces, an issue that is investigated in experiment 2.

3. Experiment 2

In experiment 1, a certain icon was always found in the same column and row. Because of the large number of trials, it is possible that the position of the individual icons was therefore highly learned. After a certain number of trials, the subjects knew the location and/or the category of an icon. Therefore, the performance measured in this way probably reflected the behaviour of practiced users.

In everyday applications, this does not have to be the case. Often, comparable interfaces are used only occasionally, so that the user cannot develop a mental model on the concrete (spatial) positions of an icon (Johnson-Laird 1983; cf. the VSSP of Baddeley 1986). Therefore, experiment 2 varied the positions of the icons from trial to trial by randomizing the locations of the columns as well as the icons within the columns. The classification of the icons to their category remained constant. The goal was to simulate occasional use;

although one has some knowledge about the interface, and may even know of all the functions, one knows nothing or very little about how to activate them.

Accordingly we anticipated that the selection times would increase on the *Q* interface, since a directed knowledge-based search for the icon within the 5×5 matrix would be more difficult. The uncertainty about the spatial position of the icon would necessitate a switch to categorical selection behaviour, so that hardly any differences are anticipated between *T* and *Q*. In contrast, the selection behaviour on the *i* should remain largely uninfluenced by this manipulation.

3.1. Method

3.1.1. *Apparatus and icons*: In contrast to experiment 1, the positions of the columns with their mode keys and the positions of the icon keys within the columns were randomized from trial to trial; otherwise the display layout was the same.

3.1.2. *Procedure and design*: These were the same as in experiment 1.

3.1.3. *Subjects*: Six female and six male college students at the University of Munich were paid to participate in the experiment. Their mean age was 29.8 years.

3.2. Results

3.2.1. *Total time*: An analysis of the total time subjects needed to solve their tasks with each interface type produced the following picture: the subjects mastered their tasks in the *i* condition with a mean of 3,827 ms per icon to be selected; in the *T* condition 3,567 ms, and in the *Q* condition 3,120 ms. Although this meant that the difference between the interface types was smaller compared with experiment 1, it continued to be significant in the ANOVA, $F(2, 22) = 33.83, p < 0.001$, as well as in the individual comparisons ($\text{Diff}_{\text{crit}, 5\%} = 177$ ms). The detailed time and error analysis brought the following results.

Task length 1. There is now significant reduction across sessions in both the start times, $F(2, 22) = 6.94, p = 0.005$, and the icon times, $F(2, 22) = 7.50, p = 0.005$. As to the differences between the interfaces, there continued to be a significant effect on icon times, $F(2, 22) = 28.07, p < 0.001$ (see figure 4). Individual comparisons showed that the mean reactions to the *i* interface (1,345 ms) were faster than reactions to *T* (2,068 ms) and *Q* (2,178 ms, $\text{Diff}_{\text{crit}, 5\%} = 317$ ms). There was no interaction between session and interface types ($p > 0.10$). A comparison of these findings with the data in experiment 1 showed that the icon times in

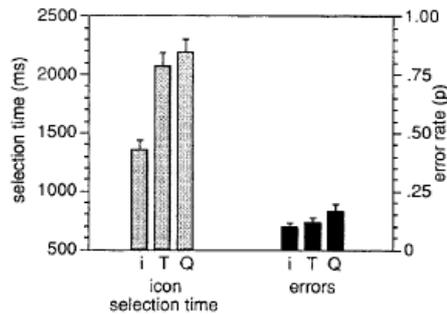


Figure 4. Icon selection times (left ordinate) and error rates (right ordinate) in task length 1 (experiment 2).

the *i* condition were hardly changed, whereas they were clearly longer in the conditions *T* and *Q*.

In the errors, there was a significant interaction between the interfaces and the session factor, $F(4, 44) = 4.03$, $p = 0.017$. While there was an increase in errors across sessions in *i* (from 0.04 to 0.13), they declined in both *Q* (from 0.27 to 0.11) and *T* (0.14 to 0.07). Errors were mostly due to unnecessarily touching the already-activated mode key (75.6% of the errors on *i*, 59.5% on *Q*, 54.0% on *T*).

Task length 2. Subjects became increasingly faster across the three sessions on all three dependent variables for task length 2: for the start times $F(2, 22) = 8.48$, $p = 0.004$; for the mode times $F(2, 22) = 21.51$, $p < 0.001$; and for the icon times $F(2, 22) = 14.42$, $p < 0.001$. There were also significant differences between the interfaces in both mode times, $F(2, 22) = 62.83$, $p < 0.001$, and icon times, $F(2, 22) = 77.24$, $p < 0.001$. In contrast with experiment 1 *T* and *Q* revealed much longer mode-selection times than *i* (see figure 5). The icon times, in contrast, were similar to those in experimental 1: for *i* they were worse than for *T*, and this, in turn, worse than *Q*. However, in experiment 2, icon times showed an interaction with the session factor, $F(4, 44) = 3.94$, $p = 0.04$. The differences between the interface types decreased at the end of the experiment although they were still present ($\text{Diff}_{\text{crit}, 5\%} = 176$ ms).

As in experiment 1, most errors were produced on the *i* interface (0.14; 86.1% of which was overshooting), and there were far fewer errors on *T* (0.07) and *Q* (0.06). The difference from the latter ones was significant, $F(2, 22) = 6.31$, $p = 0.023$, $\text{Diff}_{\text{crit}, 5\%} = 0.034$.

3.3. Discussion

The results of experiment 2 can be summarized as follows. As in experiment 1, subjects require more total time to process their tasks when facing the *i*-type interface compared with the other two interfaces. However, conditions *T* and *Q* suffer much more from the manipulations

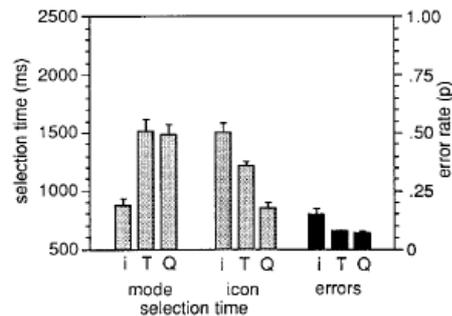


Figure 5. Mode and icon selection times (left ordinate) and error rates (right ordinate) in task length 2 (experiment 2).

in experiment 2 than the condition *i*. This is shown, first, by a clear increase in icon times in *T* and *Q* for task length 1. Second, in task length 2 selecting the correct mode key in *T* and *Q* takes even longer in condition *i*. From these results the following behavioural strategies can be derived.

As expected, the increase in selection times in *Q* and *T* owes much to the greater search effort compared with experiment 1. In *Q*, the relevant icon can be located either by a search through all 25 icons or by a categorical selection, that is, one first looks for the category (and thus the column), and subsequently for the target icon key. In *T*, it is also necessary to find the randomly positioned category first, and then the randomly positioned icon key. The extremely small difference between *Q* and *T* in the icon times of task length 1 and in the mode times of task length 2 suggests that basically the categorical selection procedure is applied to both interfaces.

In *i*, in contrast, the task is only slightly changed from experiment 1. As anticipated, the performance on *i* does not suffer from the random positioning of the icons. Searching for the icons in an item column containing only five icons continues to be fairly easy in task length 1. In task length 2, the mode and icon times once more suggest that selections in this condition can hardly be planned in advance; short mode times and relatively long icon times do not indicate the use of a categorical, but of a browsing strategy.

4. General discussion

What conclusions can be drawn from the results of these two experiments? To return to the question raised in the introduction, what can we say about the usefulness of interfaces with mode keys?

Total time. If we look at the total times subjects needed to complete their tasks, it has to be stated that the mode key impaired performance in both experiments: the *i* interface proved to be disadvantageous compared with the *T* interface

(and this, in turn, compared with the *Q* interface). When occasional use is tested (as simulated in experiment 2), the difference between the interfaces is less, but still present.

This ranking could lead to the conclusion that the mode keys are not advisable. And, in fact, such an interface should not be chosen when the time required to complete a task is the only critical variable. However, increased times does not necessarily mean that the surface is 'user unfriendly'. The analysis shows that the long times on the *i* interface are mostly due to pure motor actions, because the mode key often has to be touched several times. It is a subjective question how far such iterative motor actions are user unfriendly.

Detailed analysis of selection times. However, the two experiments also permit conclusions on the usefulness under comparable motor demands (task length 1 and 2). What advantages and disadvantages do the interfaces hold in this respect?

It is characteristic of the typical mode-key interface (*i* interface) that fewer actions can be planned in advance. The user can only check whether the desired function is present on the visible surface. If it is, it can be selected directly. Compared with the two other interfaces, this action requires little time, the low number of visible icons is effective. If the desired function is not visible, the user has to press the mode key until the desired function is available. Categorical information and spatial position play a subordinate role, as a comparison of the two experiments shows. In contrast to the two other interfaces, subjects cannot gain any great advantage from the circumstance that the interfaces are overlearned in experiment 1. Hence, a mode-key interface is closely linked to the strategy of browsing. If the user has no basic knowledge of the functions to be selected and their categorical assignments, and if the time needed to activate the relevant function is uncritical, an interface with a mode key may well be a good alternative.

What is the situation with an interface where every function is assigned its own key (*Q* interface)? Here, many actions can be planned in advance. As all functions are visible, one can search for the target and go to it directly. When the user is highly familiar with the spatial and categorical arrangement of the interface (experiment 1), planning is advantageous and can reduce the search procedure to a minimum. However, experiment 2 shows that (simulated) unfamiliar users are unable to do much forward planning, so that the great number of keys on this interface impair performance. Therefore, a type *Q* interface is recommended when the instrument is used frequently so that the user can develop a model of the spatial positions and the categorical assignment of the functions. Under these conditions, even an interface with 30 keys – as in our experiment – does not lead to any great difficulties.

The *T* interface takes an intermediate position between the other two with regard to the number of keys. However, it combines their disadvantages rather than their advantages. One cannot apply the strategy of browsing, because there is not *one* mode key but *five* categorical keys. Therefore, the user has to be very familiar with the categorical assignment of the functions to be able to select a particular function. In addition, target actions can hardly be planned in advance, because only a few function keys are visible. Hence, although the *T* interface provides a major reduction in the number of keys, this reduction makes it absolutely necessary to have categorical knowledge – in contrast with the typical *i* mode-key surface.

Because of these differential aspects, an interpretation of the present results in terms of one general mechanism could only be misleading. Fitts' law, for example, postulates a dependence of the movement time on key distance and key width (on the 'difficulty index', see Fitts 1954, Fitts and Peterson, 1964; for overviews see Rosenbaum and Krist, 1996, Müsseler *et al.* 1996). Thus, the present movement times are probably also affected by Fitts' law, but this alone is not sufficient to interpret the data. For example, in both experiments the mean movement distance for the icon-key selection is independent of task length; nevertheless, the different mode-key interfaces showed a completely diverging pattern of results when the icon times of task length 1 were compared with the icon times of task length 2. This points to other mechanisms crucial in this context.

Another more sophisticated account is the so-called 'keystroke model' (Card *et al.* 1980, 1983; for overviews see Reitman-Olson and Olson 1990). This model contains elementary operations that describe the action sequence users make in invoking simple commands (e.g. the times to mentally prepare the command, to move the hand from the mouse to the keyboard, the time to make a keystroke). However, most of these operations are kept constant between our conditions. Further, the keystroke model makes some assumptions that are hard to fulfil (i.e. serial processing of the operations, their independence, error-handling, experts' handling etc.; for a critical view see Wandmacher 1993, pp. 124–134). In the present context the results cast doubt on the serial processing of touching the mode key and then of touching the icon key. It is more plausible to assume, for example, that in the *Q* condition the target specification of the subsequent icon key is prepared *whilst* touching the mode key. Otherwise the selection times would not be so reduced. Another point is that we found differential aspects for occasional users and experts. At least in this respect the keystroke model has to be modified to fit the data.

To summarize, we can state that the present study points to differential aspects in using the different mode-key interfaces. If an instrument is designed for occasional users,

it is certainly worth considering a less complex mode-key interface. However, if most of the users will be experts, an interface where each function has its own key should be preferred.

This outcome contradicts studies by Roberts and Moran (1983) and Whiteside *et al.* (1985) in which experts and novices exhibited the same ranking in performances on various user interfaces. There could be several reasons for this: first, the interfaces tested in these studies differed to a greater extent than ours (e.g. command-line interfaces vs menu systems vs iconic systems). Second, these studies observed general performance scores (e.g. total time spent on task in minutes, percentage of task completed) whereas our study split performance into differential steps. Maybe the differential aspect becomes particularly clear at this level of a detailed analysis. Against this background, it seems to be worth registering and evaluating detailed analyses of the course of action alongside general performance scores.

Acknowledgements

We wish to thank Gisa Aschersleben and Ahmet Çakir for helpful comments on an earlier version, Jonathan Harrow and Heidi John for revising the language, and Silvia Hieke for collecting the data. Requests for reprints should be sent to either author.

Notes

¹A car cockpit was selected, since it has a sufficient large and well-known set of pictograms. Our selection, however, does not imply that we recommend a cockpit design of this kind!

²In order to avoid the risk of violating statistical assumptions, which is present in repeated-measure designs due to inhomogeneity of the variance-covariance matrix, *F* probabilities in the present and following design were corrected according to Geisser and Greenhouse (1958).

References

AREND, U., MUTHIG, K. P. and WANDMACHER, J. 1987, Evidence for global feature superiority in menu selection by icons, *Behaviour and Information Technology*, **6**, 411–426.
BADDELEY, A. 1986, *Working memory* (Clarendon Press, Oxford).
BECKER, G. 1990, Der Bildschirm im Leitstand—das ideale Interface? [The monitor in a control room—the ideal interface], in G. Daimler and K. Benz-Stiftung (eds) *Internationales Kolloquium Leitwarten 2*, (TÜV Rheinland, Cologne), 23–37.

CARD, S. K., MORAN, T. P. and NEWELL, A. 1980, The keystroke level model for user performance with interactive systems, *Communications of the ACM*, **23**, 396–410.
CARD, S. K., MORAN, T. P. and NEWELL, A. 1983, *The psychology of human-computer interaction* (Erlbaum, Hillsdale, NJ).
DIN Deutsches Institut für Normung e.V. (ed.) 1988, *Graphische Symbole nach DIN 30,600* (Beuth, Berlin).
EASON, K. D. 1984, Towards the experimental study of usability, *Behaviour and Information Technology*, **3**, 133–143.
FITTS, P. M. 1954, The information capacity of the human motor system in controlling amplitude of movement, *Journal of Experimental Psychology*, **47**, 381–391.
FITTS, P. M. and PETERSON, J. R. 1964, Information capacity of discrete motor responses, *Journal of Experimental Psychology*, **67**, 103–112.
GEISSER, S. and GREENHOUSE, S. W. 1958, An extensive of Box's results on the use of F-distribution in multivariate analysis, *Annals of Mathematical Statistics*, **29**, 885–891.
JOHNSON-LAIRD, P. N. 1983, *Mental Models* (Cambridge University Press, Cambridge).
KIGER, J. I. 1984, The depth/breadth trade-off in the design of menu-driven user-interfaces, *International Journal of Man-Machine Studies*, **20**, 201–213.
LANDAUER, T. K. and NACHBAR, D. W. 1985, Selection from alphabetic and numeric menu trees using a touch screen: breadth, depth, and width, *Proceedings of the Chi-Conference, Human Factors in Computing System* (ACM SIGCHI, New York), 73–78.
MCDONALD, J. E., STONE, J. D. and LIEBELT, L. S. 1983, Searching for items in menus: the effects of organization and type of target, *Proceedings of the 27th Annual Meeting of the Human Factors Society*, Santa Monica, CA: Human Factors Society, 834–837.
MCNAMARA, T. P. 1986, Mental representations of spatial relations, *Cognitive Psychology*, **18**, 87–121.
MÜSSELER, J. 1994, Using predictors to partition menu selection times, *Behaviour and Information Technology*, **13**, 362–372.
MÜSSELER, J., ASCHERSLEBEN, G. and PRINZ, W. 1996, Die Steuerung von Handlungen [Action control], in *Kopf-Arbeit. Gehirnfunktionen und kognitive Leistungen*, G. Roth and W. Prinz (eds.), (Spektrum, Heidelberg), in press.
PAAP, K. R. and ROSKE-HOFSTRAND, R. J. 1988, Design of menus, in *Handbook of Human-Computer Interaction*, (M. Helander (ed.) North Holland, Amsterdam), 205–235.
REITMAN-OLSON, J. and OLSON, G. M. 1990, The growth of cognitive modeling in human-computer interaction since GOMS, *Human-Computer Interaction*, **5**, 221–265.
ROSENBAUM, D. A. and KRIST, H. 1996, Preparing of movements, in *Psychomotoric*, H. Heuer, and S. W. Keele, (eds), (Academic Press, New York).
ROBERTS, T. L. and MORAN, T. P. 1983, The evaluation of text editors: methodology and empirical results, *Communications of the ACM*, **26**, 265–283.
SHERIDAN, T. B. 1988, Task allocation and supervisory control, in *Handbook of Human-Computer Interaction*, M. Helander (ed.), (North Holland, Amsterdam), 159–174.
SHINAR, D. and STERN, H. I. 1987, Alternative option selection methods in menu-driven computer programs, *Human Factors*, **29**, 453–459.
TOLMAN, E. C. 1948, Cognitive maps in rats and man, *Psychological Review*, **55**, 189–208.
TREISMAN, A. and GELADE, G. 1980, A feature-integration theory of attention, *Cognitive Psychology*, **12**, 97–136.
TULLIS, T. S. 1985, Designing a menu-based interface to an

- operating system. *Proceedings of the Chi-Conference, Human Factors in Computing System* (ACM SIGHI, New York), 79–84.
- WANDMACHER, J. 1993, *Software-Ergonomie* [Software ergonomics] (DeGruyter, Berlin).
- WHITESIDE, C. D., JONES, S., LEVY, P. S. and WIXON, D. 1985, User performance with command, menu and iconic interfaces, *Proceedings of the Chi-Conference, Human Factors in Computing System* (ACM SIGHI, New York), 185–191.