

# Direct Activation: A Concept to Encourage Tailoring Activities

**Volker Wulf and Björn Golombek**

ProSEC – Department for Computer Science III

University of Bonn

Römerstr. 164, 53111 Bonn

Germany

Tel: 1-301-403-4333

E-mail: {volker, bg}@informatik.uni-bonn.de

## ABSTRACT

The design of the user interface plays a major role in encouraging users to tailor an application. In this paper, we focus on a particular design issue. The question is how to support users in finding those functions, which allow to tailor an application. An empirical investigation shows that this is a major problem when users try to tailor applications. In order to tackle this problem we develop the concept of direct activation, which simplifies to find a tailoring function at the moment a tailorable function needs to be modified. To evaluate the effectiveness of the concept of direct activation in supporting tailoring activities, we have implemented the concept and carried out an evaluation study. The results of this study support our assumption that direct activation eases tailoring activities. Finally, the potentials and limitations of this concept are discussed.

## KEYWORDS

Tailorability, User Interface, Direct Manipulation, Field Study, Empirical Evaluation

## 1 INTRODUCTION

Tailorability is widely assumed to be a key requirement for the design of most interactive applications. During runtime it allows users to adapt an application to different tasks, personal preferences and group standards (cf. Trigg et al. 1987; Malone, Lai and Fry 1992; Paetau 1993; Oberquelle 1994; Kahler 1995; Wulf and Rohde 1995, Bentley and Dourish 1995; Syri 1997).

Tailoring can be distinguished from ordinary use and system development. Henderson and Kyng (1991) mention the following three criteria to distinguish tailoring from normal use. First tailoring modifies those aspects of an application, which are stable when performing the users' primary task. Second, tailoring activities modify rather the tool (functions of the application) than the data necessary to work on the primary task. Third, tailoring activities have longer lasting effects on the modified functions. So, the distinction between tailoring and use is relative to the users' primary task. The creation of a document template is a tailoring activity for ordinary users of a word processor. However, the same activity may be regarded as normal use for a system administrator whose primary task is the generation

of document templates. Henderson and Kyng (1991) have also differentiated between tailoring and programming. Tailoring does not lead to a new system version but modifies a given one according to local necessities. While these activities are distinct, Stiernerling et al. (1997) discuss how the realization of tailorability becomes an issue for programming activities.

With regard to the complexity of tailoring activities, Henderson and Kyng (1991) distinguish three levels:

- choice between alternatives of anticipated behavior,
- construction of new behaviors from existing pieces,
- modifying the artifact (e.g. programming).

By the following we will mainly focus on the first two levels of tailoring complexity, because they are most relevant for the key issues of this paper.

Tailorable systems can also be distinguished from adaptive systems (cf. Schneider-Hufschmidt et al. 1993; Kühme 1993; Oppermann 1994; Brusilovsky et al. 1998). While tailorable systems allow the user to keep full control over the process of modifying the application, adaptive systems build up a user model. Based on such model, they modify the system behavior automatically. Kühme et al. (1992) distinguish four levels up to which the system's behavior may be automated:

- initiative to start modifying a certain function,
- proposal of possible modifications,
- decision to chose a specific modification,
- execution of a selected modification.

It seems doubtful, however, that an automatism based on a user model shall be able to anticipate the users' requirements well enough to generate appropriate modifications on either of these levels (cf. Friedrich 1989). Therefore, this paper focuses on concepts to increase the user's control over the tailoring process by improving the user interface.

Tailorability is of specific importance for generic applications like word processors, CAD systems, e-mail tools, or workflow management systems. In these cases, a single application is supposed to satisfy the requirements of a wide variety of different users, tasks and organizational embedments (cf. Stiernerling et al. 1997). In order to adapt these applications in the desired manner, users are equipped with different tailoring functions. Tailoring functions extend or modify the functionality of an application by creating or modifying persistent artifacts. With regard to Henderson and Kyng's (1991) classification, tailoring functions typically allow to chose between alternatives of anticipated behavior or

construct new behavior from existing pieces. Together with the ordinary functionality, tailoring functions shape the user interface of an application.

We distinguish between normal and triggered function (cf. Oppermann and Simm 1994, Wulf 2000). Normal functions are typically represented at the surface interface of an application, activated by the user's input at one of their access points. They are then executed in the way prespecified by the tailoring function or specified during their activation. Focusing on graphical user interfaces according to the principle of direct manipulation, a normal function can be typically activated at the location where it is displayed on the screen (e.g. in case of menu item, button, or icon).<sup>1</sup> Nevertheless, there might exist other access points, which are not represented on the screen (e.g. short cuts). Triggered functions are not represented at the surface interface. They are activated by the occurrence of prespecified events which are either created internally or result from the interaction with external applications. A triggered function is executed in the way prespecified by the tailoring function. The deterministic relationship between the triggered and the tailoring function distinguishes a triggered function from automatism in adaptive systems. An example of a normal function is "sending mail". The "mail filter" is a function triggered by an event, the incoming mail. The function allowing users to set up the mail filter is a tailoring function. If the mail filter acted autonomously based on an automatically generated model of the user's filtering intentions, it would be called an adaptive feature.

In order to develop new concepts for the user interface of tailorable applications, we have to investigate empirically which triggers and barriers influence the usage of existing ones. Most of the fundamental research in this area has been carried out at the end of the 80s or the beginning of the 90s. The authors investigated users' tailoring habits concerning applications like window managers, word processors, spreadsheets, or CAD systems in a qualitative way (Mackay 1990; Gantt and Nardi 1992; Nardi and Miller 1991; Nardi 1993; Oppermann and Simm 1994).

Nevertheless, since that time the (tailoring) functionality of these generic applications evolved and innovative techniques for the design of the user interface spread out. Moreover, users nowadays have more experience with different types and versions of tailorable applications. Nevertheless, new technical possibilities and increased experience of users do not necessary lead to a more intense usage of tailoring functions. For instance, Palen (1997) reports that tailorable features of shared electronic calendars are very often used in the way they had been preconfigured by the providers. After all tailorability still seems to be an important research issue (e.g. Kahler et al. 2000).

---

<sup>1</sup> Rauterberg (1995) calls these locations functional access points.

In this paper we are going to focus on one particular problem concerning the design of tailorable applications. The question is how to support users in finding tailoring functions. Assuming that the functionality of generic applications is extended from version to version, this problem gains in importance. In the next section we will revise the results of earlier empirical investigations and discuss those design concepts, which have a potential to tackle this problem. Section 3 presents the results of a field study which explores problems users have in finding a tailoring function. The results of this study have motivated the concept of direct activation. This concept supports users in finding tailoring functions at the appropriate situation of use. It will be described in section 4. Prototypical implementations of the concept will be presented. To evaluate our assumptions concerning the positive effects of the concept of direct activation, we have carried out a laboratory study. These results are presented in section 5. Finally, we are going to discuss general findings.

## **2 STATE OF THE ART**

Facing the problem on how to support users in finding tailoring functions, we have to analyze at which moment they are needed. Having carried out research over a period of four months, Mackay (1990, pp. 157) describes the temporal distribution of three different tailoring activities. ‘Making new or exploratory customizations’ happens most likely just after people have received a new application. In this case tailoring activities are instrumental in learning about the functionality of a new application. ‘Changing key bindings’ happens after the introduction of a new application. It is repeated after the introduction of each new system version (retrofitting) to make the system behave in the usual way. In contrast, the tailoring activity ‘encoding repeated patterns of behavior’ is not triggered by system changes but by the actual needs of the users. Considering the importance of the different activities, Mackay (1990, p. 4) draws the conclusion: ”over time, most users make fewer and fewer customizations, regardless of level of technical expertise.” Referring to the tailoring of groupware, Tyre and Orlikowski (1994) argue even that there is only a limited timeframe in which modification to an application can opportunistically be made. According to these findings, the introduction of a new system or a new version of a system is the main occasion for tailoring.

In contrast, Page et al. (1996) come up with rather different conclusions. Recording the tailoring activities of 101 users of the word processor WordPerfect (V 6.0) during a period of 28 days, the authors found that 92% of the users tailored the system during that period. Moreover, there was a significant correlation between the general usage of ordinary functions and tailoring. Based on these findings they concluded that the current work needs are the driving force for tailoring an application (cf. Page et al. 1996, pp. 344). As Page et al. (1996) did not consider the introduction phase of the application, these

contradictive conclusions may partly be explained by the different research methods.

During empirical studies, problems to find tailoring functions have not yet been identified to be a major barrier. Nevertheless, Mackay (1990, p. 165) mentioned poor documentation and lacking knowledge about the possibilities to be factors hindering tailoring activities in a Unix environment. Oppermann and Simm (1994) found that tailoring functions were not well enough presented at the interface of the office applications they studied.

To support users in finding tailoring functions, Kragstaeter, Oppermann and Thomas (1994) have implemented an adaptive component. Based on a user model, it generates automatically recommendations for possible tailoring activities. These recommendations are displayed at the user's interface and may be accepted or rejected. Though users get aware of more tailoring functions, it is questionable whether such recommendations will anticipate the desired tailoring activity and choose the appropriate moment for its presentation. Mørch (1997) has developed a rather differed approach. In a drawing application users are able to tailor certain functions by specifying parameters, modifying design rationales and extending the source code. By pressing the "option", "shift", or "control" button while activating one of these functions, users can access the different tailoring functions. Thus, all the different tailoring functions can be activated from the same point in the user interface as the corresponding tailorable function. Applying this convention consistently to the design of the whole application, users are supported to find the respective tailoring function.

Concerning the orientation within the ordinary functionality, the visualization of functions via their functional representation points on the screen (e.g. menu items, icons, and buttons) has turned out to be helpful (cf. MacLean et al. 1990, p. 178). Moreover, survey functions, which list the existing functions according to different classification schemes, have been suggested to promote orientation (cf. Paul 1994). Nevertheless, none of these principles has yet been applied to the particular problem of finding tailoring functions at the right moment.

### **3. EMPIRICAL PRESTUDY**

In the following we are going to present the results of an empirical prestudy. We were interested in the tailoring habits of word processor users. The study aimed at identifying barriers to the use of tailoring functions. We also wanted to investigate the users' collaborative tailoring practices. The results of this explorative investigation motivated the concept of direct activation which is presented in the next section.

Word processors are generic applications, which are developed for a vast variety of different tasks and users. During the last decade the amount of their ordinary as well as their tailoring functions has increased considerably. In view of the tailoring functions, word processors nowadays allow, for instance, to create or modify document templates, to select between many different options, to modify menus and button bars, and to record or build macros. The functions "create a document template" and "modify a document template" and some of the option setting functions (e.g. select replacement mode when inserting characters) tailor normal functions. In the first two cases they tailor the function "create a new document" and in the third case the function "insert characters". Triggered functions are tailored by other option-setting functions (e.g. select language for spell checker) and the function to modify menus or button bars. In the first case the spell checker is triggered by given system states (internal events) and executed automatically according to the tailored specification. In the other two cases users' input at certain system states trigger the prespecified presentation of menus and button bars. The tailoring functions "record macro" and "build macro" do not modify existing functions but create new ones.

### **3.1 Methodology**

We conducted semi-structured interviews with eleven users of the Microsoft Word (Version 6 and Office 97) word processor in autumn 1997. We chose the employees heuristically trying to represent a variety of different levels of qualification and tasks supported by the word processor (cf. Patton 1990). Four of them were working in a federal ministry (two section heads and two administrative staff: one of the registrar and one of the mailing office). Two interviewees were working as scientists and two as UNIX system support in a research center. Finally, two interviewees studied law at the university and one was working in the marketing department of a car manufacturer.

The interviews took between 20 and 120 minutes each with an average of about 45 minutes. They were conducted at the interviewees' workplace. So, it was possible to clarify certain discussion points at the interviewees' computer screen. The interviews covered the following issues: tasks and organizational embedment of the interviewee, usage of the word processor, knowledge about ordinary and tailoring functions, ways to learn ordinary and tailoring functions, occasions for tailoring, problems in activating tailoring functions, and patterns of cooperation when tailoring a system. The interviews were concluded by a discussion about possible improvements of the tailorable aspects of the application.

We asked the interviewees to allow audio recording of the interviews and took additional notes during the interviews. All except one interviewee agreed on the recording. The interviews got transcribed and analyzed (cf. Wulf 1999a).

### 3.2 Results

Below we are going to focus on findings concerning occasions and barriers for tailoring.

#### **Tailoring: Driven by the Introduction or Emergent Needs?**

Most but not all of the interviewees tailored their application. Concerning the occasions to tailor, they mentioned the two occasions already discussed in the literature:

- after receiving a new version of the application,
- when facing emergent needs.

Like Mackay (1991, p. 157) we found retrofitting being an important phenomenon. Users tend to adapt the interfaces of a new version in a way that it looks and behaves like their old tailored version. One of the scientists describes such tailoring activities when changing from an English language Macintosh version to a German language PC version of the same word processor. "When the new version was installed I did not want to give up my Mac habits. Thus, I have made everything look like it was before. I have even created the shortcuts of the English version." Moreover, initial adaptations are often guided by the scope of tasks, which are supposed to be supported by the application. The user from the marketing division of the car producer mentioned: "I would always start to adapt the icons [button bar] because I feel bothered by all the symbols which I do not need." Thus, besides retrofitting of a certain look and feel, the anticipated scope of tasks trigger tailoring during the introduction phase.

After the intense usage of tailoring functions right after the introduction, most of the interviewees reported that unanticipated needs emerged. Tailoring is often required when new tasks arise and their performance could be better supported with a tailored application. In these situations, users often weigh the possible benefits against the efforts necessary. The user from the marketing division phrased it this way: "I do not always tailor. Sometimes I have to produce four documents in the same way. In those cases I do it manually, because tailoring is not worth the effort. In contrast, when I have to produce 30 or 40 documents it is always worth to tailor. Especially if I can use it in the future and have less work there."

The frequency of tailoring varied considerably among the different users. The interviewees carried out these activities between "once a week" and "not at all". The frequency seems to depend mainly on the individual tailoring abilities and the dynamics of their computer supported tasks.

### **Problems in Finding: Three Different Cases**

Problems to find tailoring functions in the menu were reported in almost all interviews. We can distinguish three cases:

- users already know the tailoring function,
- users imagine the existence of a tailoring function,
- users do not know nor imagine the tailoring function.

In the first case, the problem is mainly to remember the location of the tailoring function in the menu. This type of problem exists concerning ordinary functions, too. Nevertheless, due to the fact that tailoring functions are typically used less frequently, in this case the problem is even more salient. One of the administrative staff of the government organization phrased it this way: "There is a lot to be improved. Things, which you have not used for quite some time, are hard to find. You just search under the wrong name [menu item]." Our results indicate that there are problems in finding rarely needed functions if their access points are only presented in the menu.

In the second case, users imagine that a certain tailoring function should exist but they did not use it yet. Their assumptions were typically based on the following considerations:

- they knew the function in an earlier version of the application,
- they concluded by analogy that a certain tailoring function should exist,
- they had observed the effects of this function at another user's desktop,
- they had listened to other users' reports about this function.

Concerning the first bullet point, one of the scientists reported that he had not yet found the functions "tailoring of a button bar" and "recording of a macro" since he switched from the Macintosh's English version to the PC's German version. Nevertheless, he assumed that these functions existed because he knew them from the earlier version. Concerning the second bullet point another scientist mentioned that since months of usage he had not yet found the button to switch off the electronic assistant. Nevertheless, based on his experience he assumed that such a function should exist. Concerning the third bullet point, one of the Unix administrators reported how he had learned about the possibility to tailor the Word button bar: "I went somewhere [office of a colleague] and saw the bar. It was changed. I asked which version it is. He told me that it is tailorable". Concerning the fourth bullet point, interviewees reported that they had learned implicitly about tailoring functions when they had been talking to other users about related topics.



Concerning all four bullet points, the users reported about severe problems in finding the respective function. Moreover, the interviewees reported that problems to find out whether a certain function exists and how to find it had increased. Since MS-WORD's Version 6 the electronic listing of the available functions is no more in alphabetic order but in the order of the standard menu. Thus, this structure of the survey function is not very useful because it applies the same scheme as the menu.

In the third case, the users learned during the interview for the first time about tailoring functions which seemed beneficial to them. For instance, one of the scientists learned about the existence of the "tailor button bar" function during the interview. He found this function very useful.

### **Visualization and Learning**

The interviewees reported that the visual representation of (tailoring) functions motivates them to explore the functions. One of the administrative staff phrased it this way: "I like it when there is a lot on the screen. That helps me thinking and remembering that I can do many things of which I would probably not have thought of. ... Thus, I look at the button bar, can work with the ruler or change fonts and sizes. That simply helps. If I face just the screen, I think of maybe just five modifications. When facing all the options, I think of much more possibilities. I just start to try out and look."

### **3.3 Discussion and Design Implications**

The results of this explorative study suggest that tailoring of word processors is a permanent activity of those users who tailor at all. In this respect our study supports the findings of Page et al. (1996). Tailoring may be triggered by both, the introduction of a new version of an application and needs resulting from the current situation of usage. Users' needs influence the introduction phase significantly. Retrofitting typically takes need-driven aspects of earlier tailoring activities into account. Moreover, users consider future tasks during the introduction phase. So it is likely that most of the individual tailoring functions will play a role at both of these occasions.

With regard to the reasons why users search for yet unknown tailoring functions, two results of the study are interesting. First, increased experience with earlier versions or other applications can trigger tailoring. This experience becomes an important asset on which one can draw nowadays when designing tailorable systems. Second, weak modes of cooperation encourage tailoring. Mackay (1990), Gantt and Nardi (1992) and Nardi (1993) focus on intense forms of cooperation where local experts tailor and share the artifacts with other users. Our results indicate that even seeing or hearing about other users' activities may trigger tailoring. In this case users try to tailor the application themselves. Finding the

respective tailoring function is the first task on this way.

Our explorative study indicates that finding the appropriate functions is an important barrier to tailor. It either prevents tailoring or adds significantly to its costs. One may argue that these findings are rather specific to word processors which offer an increasingly vast functionality. Nevertheless, we believe that the development of word processors is rather paradigmatic for generic applications. These applications tend to increase their functionality from version to version in order to cope with new and more specific requirements. Therefore, it is worth considering the design implications of our findings.

In case tailoring is carried out as retrofitting of a new application or a new version of an application, a survey of the given tailoring functions seems to be an appropriate means to tackle the finding problem. The users get informed about the scope of tailorability. In the design-oriented discussion following the interviews, we presented screen shots where MS-WORD's tailoring functions were accessible via a newly created menu bar called "tailoring". This idea was well perceived by the interviewees, because it compensated partly for the problems with the general survey function. The interviewees estimated it eased finding and stimulated the exploration of these functions.

In case tailoring functions have to be activated need-driven during a system's usage, a context specific representation of the functions' access points seems to be more appropriate. In such a situation the user typically knows which aspects of the application he wants to modify. So, the functional access points of the tailoring functions should be visualized closely related to those aspects of the system they refer to. This is especially the case when the activation of a tailorable function leads to dissatisfactory outcomes. In this case the respective tailoring functions should be easily accessible.

Discussing MS-Word's context specific features, only few interviewees knew the context menu. Being activated by pressing the right mouse button with regard to a certain interface object, the context menu visualizes those functions, which can be executed at this occasion. In general the interviewees found it helpful to see a selection of possible functions. Nevertheless, the majority of the users did not know about this function because they were not aware of the convention for its activation. This result might be explainable due to the time of research (autumn '97) and might have changed since then. In contrast, the visual representation of access points at the user interface was reported to trigger learning of the functions.

#### **4 DIRECT ACTIVATION: THE CONCEPT**

The problems arising during the prestudy and the design-oriented discussions with the users have

motivated us to search for technical solutions. The concept of direct activation is one possible approach to tackle the given problems.

The concept of direct activation supports finding a tailoring function when it is needed. We are restricting our attention to those tailoring functions, which modify other (tailorable) functions.<sup>2</sup> Tailoring is needed when users perceive the effects of a function's execution, which does not lead to the intended effects. In the case of normal functions users are typically still aware of the function's access point. Therefore, the access point of the tailoring function should be designed related to the one of the tailorable function. By definition, triggered functions do not have an access point. Thus, the presentation of the function's execution or the outcome of its execution are the only "hooks" perceivable for the users. Therefore, the access point of the tailoring function should be designed related to the presentation of the tailorable function's execution or the outcome of its execution.

Having discussed possible hooks for normal and triggered functions, we have to define the term "related to the hook". Considering the state of the art and the results of the empirical investigation, relatedness can be achieved in two ways. First, the visual representation of the access point of the tailoring function is placed in close proximity to the hook. Second, visual representation of the functional access point may be omitted if a consistent mode exists, which allows to deduce from the hook's attributes how to activate the tailoring function (cf. Kellogg 1987, Maaß 1995).

Concerning normal functions the visual proximity of the access point of the tailoring function can be realized as follows. In case certain parameters of the tailorable function have to be specified during the activation, visual proximity can be reached by displaying the access point of the tailoring functions next to the one for specifying the parameters (e.g. in the same window). If the tailorable function is executed without further specification from the menu or via an icon, the access point for the tailoring function could be placed next to the one of the tailorable function. Concerning triggered functions, visual proximity can be reached by integrating the functional access point into the visualization of the function's execution or the outcome of its execution.

Mørch (1997) gives an example of a consistent mode how to deduce the way the tailoring function is activated based on the attributes of the hook. The users can access tailoring functions by activating the normal function and pressing additionally specific buttons (cf. chapter 2). Restricted to certain triggered functions, the Microsoft context menu gives an idea of how to design a consistent mode to activate the respective tailoring functions. Whenever the display of a screen object results from the execution of a

---

<sup>2</sup> Tailoring functions which create new ordinary function (e.g.: record a new macro) are not covered.

tailorable function, a specific mouse operation on this object allows accessing the tailoring function.

Comparing the two modes of direct activation, the results of the prestudy seem to favor the first approach. A visual representation of the access seems to stimulate learning and usage of a function. In contrast, the context menu had not been discovered by many users yet. Therefore, we believe that direct activation should be implemented via visual representation of the functions' access points as long as the concept is not applied to at least the entire application (cf. Mørch 1997).

The tailoring functions are presented twice at the surface interface of an application in case one implements a survey function and the concept of direct activation by means of visual representation. In case a function gets activated via its own specification window, there should be no problem to add the access point of the tailoring function. In case a function is activated without further specification, the visual representation of the tailoring functions' access points has to be placed on the same level of the menu as the one of the tailorable function. As the space on the interface is limited, this may cause problems. In this case, two modes of the user interface may be implemented: a use and a tailoring mode. The graphical presentation of the access points for the tailoring functions will be placed in the proximity of the respective tailorable function just in the tailoring mode. While working typically in the ordinary mode, users mostly switched into the tailoring mode when looking for certain tailoring functions.

## 5 EMPIRICAL EVALUATION

The concept of direct activation was motivated by users' problems which arose during the empirical prestudies. We are now going to evaluate whether this concept helps to overcome the mentioned problems.<sup>3</sup> The concept of direct activation supports users in having access to tailoring functions when necessary. So users whose applications are implemented according to the concept of direct activation, should have less problems to find tailoring functions. These users should be better in finding tailoring functions, and therefore, be able to tailor in a better and faster manner than others whose applications are not implemented in this way. Moreover, we assumed that support in finding tailoring functions had a positive effect on the users' understanding of these functions. We assumed that this effect was particularly important in case direct activation is realized by visual representation of the functions' access points. These considerations lead to the following hypotheses.

*Hypothesis 1: Direct activation leads to a better finding of tailoring functions.*

*Hypothesis 2: Direct activation leads to a better tailoring performance.*

---

<sup>3</sup> This study was part of a larger experiment in which other concepts to encourage tailoring activities have been tested, too (cf. Wulf 1999b).

*Hypothesis 3: Direct activation leads to a faster tailoring performance.*

*Hypothesis 4: Direct activation leads to a better understanding of the tailoring functions.*

In order to test these hypotheses, we have developed two different versions of a tailorable awareness service for a tool which allows sending and sharing tailored artifacts (e.g. document templates or button bars) in a word processor. The first version implemented the concept of direct activation, while in the second version this concept has not been realized.

## **5.1 Technical Artifacts**

The tailorable awareness service is implemented based on a groupware tool which supports cooperative tailoring activities. The groupware tool extends the MS-Word application. The version of the tool used in the experiment allows to exchange two types of tailored artifacts: document templates and button bars. Each user of the groupware tool has a private workspace to store tailored artifacts and a mailbox to receive tailored artifacts sent by other users. Moreover, a shared workspace is provided to allow users to publish tailored artifacts from their private workspaces and copy tailored artifacts to their private workspaces. This tool has been developed in VBA (Visual Basic for Applications) and integrated in the MS-Word menu bar (cf. Kahler et al. 1999).

In order to inform users about events concerning shared tailored artifacts, an awareness service is implemented in the tool. Eight different users' activities are recorded separately for both types of tailored artifacts: document templates and button bars. So sixteen types of events can be distributed automatically among the users and displayed at their interfaces. Examples of these event types are: publishing a new button bar to the shared workspace, copying a document template from the shared to the private workspace, applying a button bar from the private workspace, deleting a document template from the mailbox. To avoid users being overloaded by receiving useless messages and to protect their privacy, the awareness service is tailorable by the realization of two filters. These filters enable the users to tailor the awareness service in a flexible way. Both sides, the producer and the recipient, are able to influence the event-flow within the system.

As there is a rather high level of flexibility implemented in the filters, the tailoring process may become complex. In order to support users in handling this complexity two layers of tailoring functions are implemented for every filter. The concept of direct activation is applied to support the users to find these tailoring functions, and thus, to ease the tailoring process.

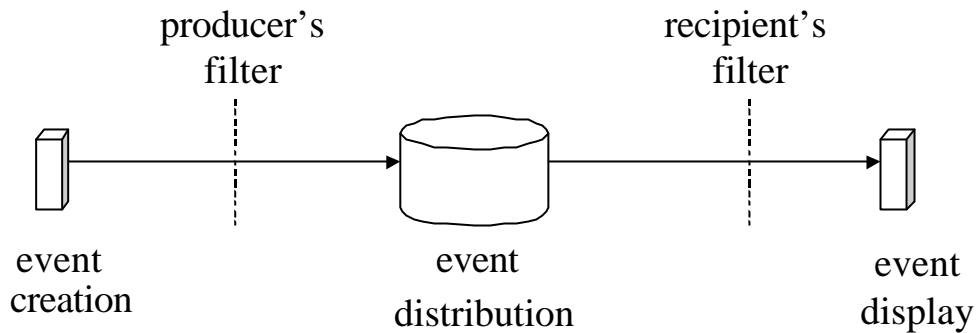


Figure 1: Architecture of the tailorable awareness service

The recipient's filter allows selecting those events which should be displayed at the users' screen. Therefore, the users are able to specify which type of event (choosing from a list of sixteen predefined event-types) should be displayed in a certain situation (choosing from a list of four predefined situations) at a given intensity (choosing from a list of three predefined intensities). The users can select among the following situations when the events are displayed at their interface: when starting the word processor, when starting the groupware tool, when opening the mailbox, or at anytime. Concerning the display of the events, the users can choose among the following display intensities: pop up window, icon presentation in the status bar, or no display at all.

To set up the recipient's filter, we provide two tailoring functions of different levels of complexity. On the first level, users can select an interest profile per event-type. The interest profiles are presented by their names in form of a list. Each name describes the profile's behavior in plain text. By selecting one of these profiles the recipient's filter is tailored. On the second level of complexity, the users can generate new interest profiles. They select one display situation and attached it to a display intensity. For such a newly created profile, a name is automatically generated in plain text. This name may be edited by the user at any time. The profiles' names are shown when a user activates a tailoring function on the first level of complexity. By means of the first level tailoring functions a profile can be selected out of the given list of profiles.

During the work with the groupware tool, the recipients' filter controls the distribution of the published events. If the user's actual situation is identical to the situation specified in the selected profile of the recipient's filter, the event will be displayed at the screen .

The producer's filter allows the user to decide which of their events should be published to whom. Here again two layers of tailoring complexity are provided. On the first level of complexity the user can select which other user or user group should be notified about the occurrence of a certain type of event. On the second level, it is possible to create new groups of users to whom the events should be distributed. These groups will be the recipients of an event if their recipient's filter allows to display the event. Users who are not specified in the producer's filter can not receive events.

We applied the concept of direct activation to support users in finding the tailoring functions which allowed them to specify the recipient's and the producer's filter. As the application was fully embedded in the MS-Word environment, it was difficult to create a consistent non-visible mode to support users in having access to the tailoring functions. Therefore, we used graphical proximity as the way to realize the concept of direct activation (cf. section 4).

Both filters are triggered functions. The effect of the execution of the recipient's filter is only visible in case events are displayed at the user interface. Therefore, we decided to use the display of the incoming events as the hook for an access to the first level tailoring function. In case the events were displayed in a pop up window, we added a button to the window which allows accessing the first level tailoring function, specifying the recipient's filter. In case they were displayed as an icon in the status bar, we placed an additional icon referring to the recipient's filter in the same bar.

The execution of the producer's filter and the outgoing events did not get visible at all. Anyway such a design poses privacy problems (cf. Fuchs 1997). Therefore, we decided to visualize the production of outgoing events by animating the functional access point of the first level tailoring function. This press button of the first level tailoring function changed its color briefly whenever a user carried out an action which created an outgoing event.

Tailoring functions are always normal functions. They are typically activated via an own window because they need to be specified by the users. Therefore, we placed the access point of the second level tailoring function into the windows activating the first level tailoring function. We extended the pull down menus where the users could select the interest profile (in case of the recipient's filter) and the groups of users (in case of the producer's filter) by an item representing the second level tailoring function. Choosing this item the users activate the window to specify the second level tailoring functions. Figure 2 shows such a realization of direct activation for the case of the producer's filter. When tailoring the producer's filter concerning the event-type "delete document templates from the private workspace", a list of users and user groups is displayed. By selecting an item out of this list, the

event is published to the corresponding group of recipients. In case the items of the list do not convene to the user's intention, he may select the "tailor" option. Via the "tailor" option he may now get access to the second level of tailoring functions to generate a new group of users.

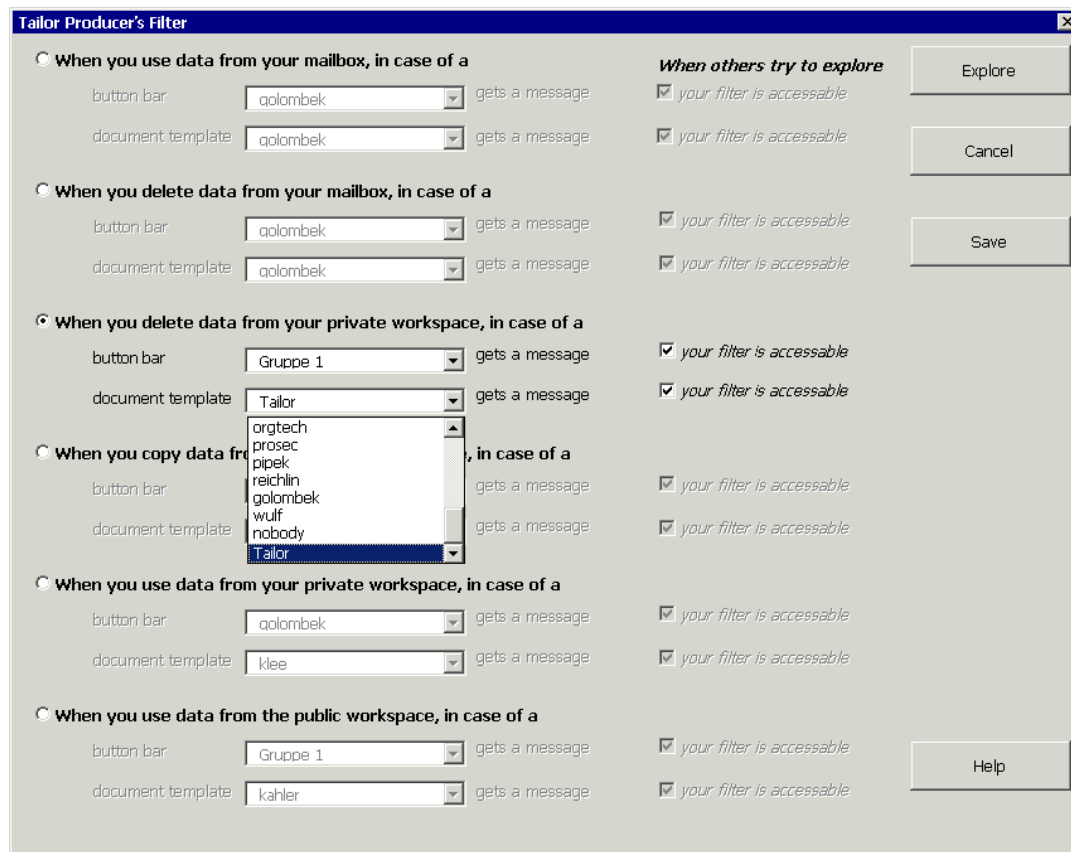


Figure 2: Direct activation of the second level tailoring function in case of the producer's filter

The second version of the tailorable awareness service was not realized according to the concept of direct activation. The access points of all tailoring functions which allow to specify the filters were placed in a newly created menu bar called “tailoring”. In this menu bar the access points of further tailoring functions were placed, as well (cf. section 3.3). By means of an experiment we were now able to compare the performance of users working with the two different versions of the tool (experimental conditions).

## 5.2 Method

Our experiment involved 22 persons, 11 for each experimental condition. Their age ranged from 20 to 35 years. About 80% of the subjects were students with a variety of specialties. Moreover, office



workers with different professional backgrounds took part in the study.

Beforehand we checked the subject's computer experience by means of a questionnaire. Based on the result of this investigation, a rather homogeneous group of subjects was selected. All of them had already used word processors before, while none had experiences as a programmer. We also documented further aspects of their computer experience (e.g. frequency and type of tailoring activities, experience with e-mail). The subjects were assigned at random to the experimental conditions. We are going to call those subjects whose application was implemented according to the concept of direct activation the DA group, the other group is called the control group. At the end of the experiments all subjects participated in a lottery with the possibility of winning two hundred German Marks.

The experiment took place in the research lab of the University of Bonn. The lab was equipped with two networked computers on which the groupware tool was running. The subjects were asked to use the first computer while the second one was used by one of the experimenters during the experiment. The subjects were informed that the same tool was running on both of the computers. During the whole experiment the subjects could move to the second computer, watch the screen, or ask the experimenter for access. So, the experimental setting allowed the users to have access to a second interface and learn about the groupware's awareness service.

The tool did not imply a help function. The subjects could ask the experimenter whenever they had questions concerning the tool. However, the experimenter did not answer questions related to the given tasks.

The test consisted of two phases which were identically for both experimental conditions. In the first phase the subjects were asked to explore the functionality of the tool while in the second phase they had to work on given tasks. The first phase of the experiment was started with an overview of the given version's functions. In order to direct the learning activities towards the functions relevant for the second phase of the test, some tentative tasks were provided for the subjects. The task performance was not checked during the first phase of the experiment. The subjects themselves decided when to finish with this phase of the experiment. The experimenter stopped the learning phase after a maximum time span of half an hour. All subjects, except for two, made full use of the given time for learning.

In the second phase of the experiment five tasks had to be solved. The performance concerning these tasks was supposed to test the hypotheses. The first two tasks asked the subjects to modify the filter settings by means of the given tailoring functions. Further three tasks asked to predict the behavior of the tailorable awareness service. We started the second phase of the experiment by reconfiguring the

awareness service in a similar way for all subjects. There was no time limitation for the second phase of the experiment. The subjects could work on the tasks as long as they wanted.

The first task was to modify the display intensity for the of a certain type of event, which had been created by one of the experimenters. Due to the preconfiguration of the awareness service, it was displayed on the subjects' screen at the intensity level of a „pop-up window“. The subjects were asked to reduce the display intensity to „icon presentation in the status bar“. The second task demanded modifying the setting of the recipient's filter, again. However, this time it was necessary to create a new interest profile. So the subjects had to activate two different tailoring functions. A new interest profile had first to be created and then selected (see appendix).

The experimenter noted whether the subjects reached the access points of all task-relevant tailoring functions and whether they carried out the tailoring tasks correctly. If all the relevant access points to the tailoring functions had been reached, the finding task was regarded to be fulfilled. We coded this fact with a one. In case they did not reach all relevant functional access points, we coded this fact with a zero. In the same way we coded the success of the tailoring activities. A one indicated that the tailoring task had been carried out successfully, a zero indicated that the given task had not been carried out successfully.

Three more tasks checked the subjects' overall understanding of the functionality. They were asked to predict the behavior of the tailorable awareness service. While tasks three and four asked the users to predict the behavior of the recipient's filter, task five dealt with the producer's filter. Task three demanded to predict whether and how one would be notified about a document template mailed to them. Task four asked the subjects to predict how recipients got aware of the fact that a tailored button bar had been sent to them. Task five required a prediction whether and how other users would become aware of the fact that they had loaded a button bar from their private folder. A predefined state of their producer's filter was given in the task description (see appendix). Figure 3 presents the description of task four.

*You send the button bar "project x" to the user "golombek". How can he perceive this fact?*

***O** not at all because .....*

***O** he finds it in his private work space because .....*

***O** he finds it in his mailbox because .....*

***O** by means of the awareness service, in case .....*

Figure 3: Description of task 4

For each of the three tasks one or more predictions had to be selected out of a list of given alternatives. In some cases, the subjects had to give a brief explanation why they had chosen a certain alternative. Depending on the difficulty to predict the behavior of the awareness service, each correctly selected alternative was counted as either one or two points. Each correct explanation was worth another one or two points. An incorrectly selected alternative or a false explanation reduced the amount of points accordingly. Following this scheme the subjects could reach a maximum of three points in the third task, a maximum of four points in the forth task and a maximum of seven points in the fifth task (cf. table 1). The results of each task were scaled down to the  $[0, 1]$ -interval later on by dividing the points reached through the maximum amount of possible points per task.

While working on these three tasks, the subjects were still sitting at their computer and could use the system to find out appropriate solutions.

Task 1	Finding: $x \in \{0, 1\}$
	Tailoring: $x \in \{0, 1\}$
Task 2	Finding: $x \in \{0, 1\}$
	Tailoring: $x \in \{0, 1\}$
Task 3	Understanding: $x = y / 3$ $y \in \{0, 1, 2, 3\}$
Task 4	Understanding: $x = y / 4$ $y \in \{0, 1, 2, 3, 4, \}$
Task 5	Understanding: $x = y / 5$ $y \in \{0, 1, 2, 3, 4, 5, 6, 7\}$

Table 1: Points to be gained per task

### 5.3 Empirical Results

In this section the results of the empirical investigation will be presented. Table 2 gives a survey of the subjects' performance concerning tasks 1 and 2 and their aggregated performance. While the scores for finding the tailoring functions are relevant to test hypothesis 1 (direct activation leads to a better finding of tailoring functions), the scores of the tailoring performance are essential to test hypothesis 2 (direct activation leads to a better tailoring performance). The aggregated performance is the sum of the performances of both tasks divided by two. While all subjects of the two groups found the tailoring functions which were relevant to carry out task 1, only a certain part of the users (82% of the DA group and 55% of the control group) carried out the tailoring task correctly. With regard to task 2 only in the DA group all the subjects found the two tailoring functions relevant for their task performance. In the control group only about half of the subjects found both tailoring functions. Looking at the task

performance, it was highest in the DA group, too (55% of the subjects were successful in the DA group vs. 9% in the control group).

Table 2 also presents the time in minutes, the subjects took for the tailoring tasks. We used the data concerning the time spent for tailoring to test hypothesis 3 (direct activation leads to a faster tailoring performance). With regard to the tailoring tasks, the DA group was faster in carrying them out than the control group.

Finding and Tailoring Performance	Tasks	Control Group		DA Group	
		m	sd	m	sd
Finding of the Tailoring Function	Task 1	1	0	1	0
	Task 2	0.55	0.52	1	0
	(Tasks 1 + 2) / 2	0.78	0.26	1	0
Tailoring Performance	Task 1	0.55	0.52	0.82	0.4
	Task 2	0.09	0.30	0.55	0.52
	(Tasks 1 + 2) / 2	0.32	0.34	0.68	0.34
Time Spent for Tailoring (in minutes)		(Tasks 1 + 2)	9.55	4.16	5.91
					1.82

Table 2: Mean values and standard deviation: finding the tailoring function, tailoring performance and the time spent for tailoring (tasks 1 and 2)

Table 3 presents the subjects' performance on tasks 3, 4 and 5 and their aggregated performance. We used these data to test hypothesis 4 (direct activation leads to a better understanding of the tailoring functions). For each of the tasks the result has been scaled down to the [0, 1]-interval by dividing the points gained by the maximum amount of points to be reached per task. The aggregated performance is the sum of the performances of the three tasks divided by three, and this scaled down to the [0, 1]-interval. The data do not show a strong difference in understanding between the DA and the control group.

Understanding	Control Group		DA Group	
	m	sd	m	sd
Task 3	0.818	0.23	0.788	0.27
Task 4	0.546	0.31	0.523	0.33
Task 5	0.701	0.34	0.623	0.38
(Tasks 3 + 4 + 5) / 3	0.69	0.23	0.65	0.22

Table 3: Scaled mean values and statistical deviation of task performance (tasks 3, 4 and 5)

In order to find out whether the results presented so far indicate significant differences, we carried out single tail t-tests between the results of the DA group and the control group. Table 4 summarizes the results of the tests carried out to validate the different hypotheses. The first two tests are based on the aggregated outcomes of tasks 1 and 2, which are presented in table 2. The results of the first ttest indicate that the DA group was better in finding the relevant tailoring functions than the control group in a highly significant way. These results support hypothesis 1 (direct activation leads to a better finding of tailoring functions). The DA group was also significantly better in solving the tailoring tasks, which supports hypothesis 2 (direct activation leads to a better tailoring performance). Interestingly, the DA group was also in a highly significant way faster in tailoring than the control group. This finding supports hypothesis 3 (direct activation leads to a faster tailoring performance). So the trends which can be already observed in table 2, support our hypotheses in a significant way. However, hypothesis 4 (direct activation leads to a better understanding of the tailoring functions) has to be rejected based on the results of the t-test (cf. table 4).

	group	m	sd	group	m	sd	df	t-value	p < 0,1
H 1	DA Gr.	1	0	C Gr.	0.78	0.26	20	2.89	0.008 <sup>++</sup>
H 2	DA Gr.	0.68	0.34	C Gr.	0.32	0.34	20	2.53	0.010 <sup>+</sup>
H 3	DA Gr.	5.91	1.82	C Gr.	9.55	4.16	20	-2.66	0.009 <sup>++</sup>
H 4	DA Gr.	0.65	0.22	C Gr.	0.69	0.23	20	-0.51	0.618

<sup>+</sup>significant (p < 0.05), <sup>++</sup>highly significant (p < 0.01)

Table 4: Results when testing the given hypotheses

Focusing more detailed on table 2, it turns out that the implementation of direct activation has special advantages concerning task 2. Obviously the concept of direct activation has the most important advantages when more complex tailoring activities have to be performed and users have to activate the second level tailoring function to carry out these tasks. These results support our general assumption about the positive effects of direct activation.

There are two possible interpretations to the better and faster tailoring performance of the DA group compared to the control group. First, the fact that users easier find the tailoring function leads to a higher efficiency of the tailoring activities as a whole. Second, the fact that users easier find the tailoring functions encourages better learning in the first phase of the experiment. Better learning in the first phase increases the users' understanding of the functionality. A better understanding in turn makes the tailoring performance more efficient. The rather low performance of the DA group concerning the task 3, 4 and 5 points to the first interpretation of the facts.

While direct activation does not lead to a better understanding of the way groupware functions work, it improves the users' orientation within highly tailorable applications. Such an increased orientation has obviously an important impact on the efficiency of the tailoring activities. So far the results of the experiment go along with the conclusions drawn from the empirical prestudy.

## **5.4 Discussion**

There are some limitations to the results presented. We just implemented the concept of direct activation concerning a tailorable awareness service of a specific groupware tool. Further investigations are needed to find out whether there is a positive effect on the users' orientation with regard to other tailorable applications, as well. Moreover, the concept of direct activation is formulated in a rather general sense. We have evaluated just one possible implementation. Nevertheless, one may think of other implementations which differ from the ones evaluated. For instance, the concept could be implemented with a consistent mode of activation instead of visualizing the access points of the tailoring functions. Further research is therefore needed to evaluate different variations of the concept. Another limitation of the results of the study stem from the selection criteria of the subjects. All of them had already some experience with word processors, which may have influenced the results of the study. Less or more qualified subjects may have had valued the concept of direct activation in a different way.

Taking these limitations into consideration, most of our assumptions are supported by the data. Direct activation seems to ease locating tailoring functions. Easier finding the tailoring functions leads to more efficient tailoring activities. So the first three hypotheses are supported by the findings of this study. However, the results of the evaluation study indicate that direct activation does not lead to a better understanding of the users with regard to the tailorable functions. This is contradictory to the findings of the prestudy in which users mentioned that the visible representation of a function' access point encourages learning (cf. section 3.2). This contradiction may be due to the fact that in our evaluation study a complex groupware functionality (tailorable awareness service) had to be understood while the prestudy dealt with functions of a single user application (word processor). It is typically much more difficult for users to build up an appropriate understanding of functions in groupware than in single user applications (cf. Wulf 1999b). The rather negative finding with regard to the concept's impact on learning may such be due to the complexity of the tailorable functions on which the experiment has been based. Further research is required at this point.

## **6 CONCLUSION**

The empirical prestudy dealing with a word processor's use indicates that tailoring is required

permanently. Tailoring is triggered by both, the introduction of a new system version and non-anticipated needs emerging during a system's use. Finding the appropriate tailoring functions is a major barrier to tailoring. It either increases the efforts necessary to activate the tailoring functions or prevents the users from recognizing the tailoring options at all.

To tackle this barrier two concepts should be implemented in parallel. Survey functions and the concept of direct activation make the tailorable aspects of an application perceivable. Survey functions inform users about the tailorable aspects of an application. Such a survey is best fitted for the introduction phase, when applications have to be retrofitted, and for the usage phase, when new functions have to be generated. Per contra, direct activation is the appropriate concept if a specific tailorable function requires attention. Such a situation may come up during the introduction phase, when exploring tailorable functions, or during the use phase, when an existing function fails due to inappropriate preconfiguration. Contrary to the menu structure in general, functional access points to tailoring functions can be allocated task-orientedly because the occasions for their activation can be anticipated. The tailoring task arises when problems concerning the respective tailorable function are perceived.

In this paper, we have focused on the concept of direct activation. It was discussed how the finding of tailoring functions may be supported either by visual representation of the access point or by a consistent mode of activation. We have presented examples how to implement the concept of direct activation. Moreover, the concept of direct activation has been evaluated. The test-statistical evaluation supports our assumption that this concept eases finding tailoring functions when they are needed, and therefore leads to better and more time-efficient tailoring activities. Our investigation has been based on word processors and their groupware-oriented extensions. However, we believe that the concept of direct activation is relevant to any generic application which provides a high degree of tailorability to its users.

The concept of direct activation applies the principles of direct manipulation to the problem of finding tailoring functions (cf. Shneiderman 1983). As the objects (tailorable functions) are linked task-orientedly to the tailoring functions, the users are more directly engaged with controlling the objects, instead of searching. Thus, less cognitive resources are required (cf. Hutchins, Hollan, and Norman 1986, pp. 91). Compared to automatic recommender systems (cf. section 2) the user stays in control and decides himself when to look for a tailoring function.

Like the paradigm of direct manipulation and other design principles (e.g.: ISO 9241, part 10), the concept of direct activation just provides qualitative guidelines. Its concrete implementation has to be worked out concerning each tailorable function and application. In this paper we have based our

consideration mainly on interfaces following the paradigm of direct manipulation. However, the guiding principles of direct activation can be applied to other interface paradigms (e.g. command language), as well. Thus the results presented here may stimulate further research on how to ease the orientation of users within highly tailorable applications.

## REFERENCES

- Bentley, R.; Dourish, P.: Medium versus Mechanism. Supporting Collaboration Through Customisation. In: Marmolin, H.; Sundblad, Y.; Schmidt, K. (Hrsg.), Proceedings of the Fourth European Conference on Computer Supported Cooperative Work - ECSCW '95, Kluwer, pp. 133-148
- Brusilovsky, P.; Kobsa, A.; Vassileva, J. (eds): Adaptive Hypertext and Hypermedia, Kluwer, Dordrecht 1998
- Friedrich, J.: Adaptivität und Adaptierbarkeit informationstechnischer Systeme in der Arbeitswelt - Zur Sozialverträglichkeit zweier Paradigmen. In: Reuter, A. (Hrsg.): Proceedings der 20 GI-Jahrestagung, Springer, Heidelberg u. a. 1990, pp. 178 - 191
- Fuchs, L.: Situationsorientierte Unterstützung von Gruppenwahrnehmung in CSCW-Systemen, PhD-Thesis, Department of Computer Science, University of Essen, 1997
- Gantt, M.; Nardi, B. A.: Gardeners and gurus: Patterns of Cooperation among CAD users, in: Proceedings of CHI '91, ACM-Press, New York, pp. 107 - 117
- Henderson, A.; Kyng M. (1991): There's No Place Like Home. Continuing Design in Use. in: Design at Work, Lawrence Erlbaum Associates, Publishers, pp. 219-240
- Hudchins, E. L.; Hollan, J. D.; Norman, D. A.: Direct Manipulation Interfaces, in: Norman, D. A.; Draper, S. W. (eds): User Centred System Design, Hillsdale, pp. 87 - 124
- ISO 9241: Ergonomic requirements for office work with visual display terminals (VDTs) Part 10: Dialogue Principles
- Kahler, H.: From Taylorism to Tailorability. In: Anzai, Y.; Ogawa, K.; Mori, H. (eds): Symbiosis of Human and Artifact, Proceedings of HCI '95, Vol. 20B, Elsevier, Amsterdam 1995, pp. 995-1000.
- Kahler, H.; Stiernerling, O.; Wulf, V.; Höpfner, G.: Gemeinsame Anpassung von Einzelplatzanwendungen, in: Proceedings of Software-Ergonomie '99, Teubner, Stuttgart, 1999, pp. 183 - 194
- Kahler, H.; Mørch, A.; Stiernerling, O.; Wulf, V.: Tailorable Systems and Cooperative Work, Special Issue of JCSCW: The Journal of Collaborative Computing, Vol. 9, No. 1, 2000
- Kellogg, W. E.: Conceptual Consistency in the user interface: Effects on user performance, in: Bullinger, H.-J.; Shaker, B.: Human Computer Interaction - INTERACT '87, Elsevier 1987, pp. 389 - 394
- Krogsaeter, M.; Oppermann, R.; Thomas, Ch. G.: A User Interface Integrating Adaptability and Adaptivity, in: Oppermann, R. (Hg.): Adaptive User Support, Hillsdale 1994
- Kühme, T.: A User-centered Approach to Adaptive Interfaces, in: Proceedings of the International Workshop on Intelligent User Interfaces (IUI '93), January 4 - 7, 1993, Orlando, FL, ACM-Press, New York 1993, pp. 243 - 245
- Kühme, T.; Dietrich, H.; Malinowski, U.; Schneider-Hufschmidt, M.: Approaches to Adaptability in User Interface Technology, in: Unger, C.; Lawson, J. A. (eds): Proceedings of IFIP-WG 2.7 Working Conference on Engineering for Human-Computer Interaction, Elsevier Amsterdam 1992, pp. 225-250
- Maaß, S.: Strömungen, Leitbilder und Begrifflichkeit der Software-Ergonomie - Beiträge zur Entwicklung und Systematisierung eines Fachgebietes, Habilitationsschrift, Universität Hamburg, 1995
- Mackay, Wendy E.: Users and customizable Software: A Co-Adaptive Phenomenon, PhD-Theses, MIT, Boston (MA) 1990



- MacLean, A.; Carter, K.; Lövsstrand, L.; Moran, T: User-tailorable Systems: Pressing the Issue with Buttons, in: Proceedings of CHI '90, ACM-Press, New York 1990, pp. 175–182
- Malone, Th. W.; Fry, Ch.; Lai, K.-Y.: Experiments with Oval: A Radically Tailorable Tool for Cooperative Work. in: Proceedings of CSCW '92, ACM-Press, New York, 1992, pp. 289-297
- Mørch, A.: Method and Tools for Tailoring of Object-oriented Applications: An Evolving Artifacts Approach, PhD-Thesis, University of Oslo, Department of Computer Science, Research Report 241, Oslo 1997
- Nardi, B. A.; Miller, J.: Twinkling lights and nested loops: Distributed problem solving and spreadsheet development, in: International Journal of Man Machine Studies, Vol 34., 1991, pp. 161 – 184
- Nardi, B. A.: A Small Matter of Programming - Perspectives on end user computing, MIT-Press, Cambridge et al. 1993
- Oberquelle, H. (1994): Situationsbedingte und benutzerorientierte Anpaßbarkeit von Groupware. in: Hartmann, A. et al. (eds), Menschengerechte Groupware, Teubner, Stuttgart, pp. 31 -50
- Oppermann, R. (ed.): Adaptive User Support, LEA, Hillsdale, 1994
- Oppermann, R.; Simm, H.: Adaptability: User-Initiated Individualization, In: Oppermann, R. (ed.): Adaptive User Support, Hillsdale, 1994
- Paetau, M.: Konfigurative Technik und die Dynamik sozialer Systeme - ein Überblick über die Gestaltungsproblematik und ein Lösungsvorschlag. In: Müller, W.; Senghaas-Knobloch, E. (Hrsg.): Arbeitsorientierte Technikbewertung und Softwaregestaltung - Leitbilder, Methoden und Werkzeuge. Münster u. a. 1993, S. 149 - 180
- Page, S.; Johnsgard, T.; Albert, U.; Allen, C.: User Customization of a Word Processor. In: Proceedings of CHI '96, April 13.-18 1996, pp. 340-346
- Palen, L.: Groupware Adoption & Adaptation: Studies on Successful Use, in: SIGGROUP Bulletin, Vol. 18, No. 3, 1997, pp. 51 - 55
- Patton, M. Q.: Qualitative Interviewing, in: Patton, M. Q. (ed): Qualitative Evaluation and Research Methods, Sage, 1990, pp. 277-359
- Paul, H.: Explorative Agieren, Frankfurt/M 1994
- Rauterberg, M.: Ein Konzept zur Quantifizierung software-ergonomischer Richtlinien, PhD-Thesis, University of Zürich, Zürich 1995
- Schneider-Hufschmidt, M.; Kühme, T.; and Malinowski, U. (eds): Adaptive User Interfaces: Principles and Practice, North Holland Elsevier, Amsterdam, 1993
- Shneiderman, B.: Direct Manipulation: A Step beyond Programming Language, IEEE Computer, Vol. 16, No. 8, 1983, pp. 57 – 68
- Stiernerling, O.; Kahler, H. and Wulf, V.: How to Make Software Softer - Designing Tailorable Applications. In Proceedings of 2<sup>nd</sup> Conference on the Design of Interactive Systems, Amsterdam (NL), ACM Press 1997, pp. 365-376
- Syri, A.: Tailoring Cooperation through Mediators, in: Hughes, J. A.; Prinz, W.; Rodden, T.; Schmidt, K. (eds): Proceedings of the Fifth European Conference on Computer Supported Cooperative Work - ECSCW '97, Kluwer, Dordrecht 1997, pp. 157 - 172
- Trigg, R. H.; Moran, T. P. and Halasz, F. G.: Adaptability and tailorability in Note Cards, in: Bullinger, H.J., Shackel, B. (eds): Human-Computer Interaction - INTERACT '87, Elsevier, Amsterdam, 1987, pp. 723 – 728
- Tyre, M. J.; Orlikowski, W. J.: Windows of Opportunity: Temporal Patterns of Technological Adaptation in Organizations, in: Organization Science, Vol. 5, No. 1, 1994, pp. 98 – 118
- Wulf, V.: On Search for Tailoring Functions: Empirical Findings and Implications for Design, in: Proceedings of OZCHI '99, November 28 - 30, 1999, Wagga Wagga, Australia, CSU-Publisher, 1999, pp. 105-111

- Wulf, V.: Design for Self-directed Learning – Evaluating the Concepts of Direct Activation and Explorative Execution, Working Paper, ProSEC – Department of Computer Science, University of Bonn, 1999b
- Wulf, V.: Exploration Environments: Supporting Users to Learn Groupware Functions, to appear in: Interacting with Computers, 2000
- Wulf V. and Rohde M.: Towards an Integrated Organization and Technology Development; in: Proceedings of DIS '95, ACM-Press, New York 1995, pp. 55 – 64

## Appendix: Task Descriptions for the Second Phase of the Experiment

### Task 1

*Right now the event-type “delete a button bar from the mailbox” is displayed with the intensity “pop-up window” on your screen! Please set up your recipient’s filter in a way that this event-type is displayed as an icon in the status bar whenever the word processor is started.*

### Task 2

*Please set up your recipient’s filter in a way that events of the type “copy a document template from the shared to the private workspace” are indicated as an icon in the status bar.*

### Task 3

*The user “golombek” sends you a document template. How can you perceive this fact?*

- ☐ an icon is displayed in the status bar because .....*
- ☐ a pop-up window is displayed because .....*
- ☐ there is not any display because .....*
- ☐ the document template will be in my private workspace*
- ☐ the document template will be in my mailbox*

### Task 4

*You send the button bar “project x” to the user “golombek”. How can he perceive this fact?*

- ☐ not at all because .....*
- ☐ he finds it in his private workspace because .....*
- ☐ he finds it in his mailbox because .....*
- ☐ by means of the awareness service, in case .....*

### Task 5

*You have chosen the user group “usability” when setting up your producer’s filter for the event type “apply button bar from the private workspace”. What happens when you use a button bar from your private workspace?*

- ☐ the user “andiel” does not perceive the event because ....*
- ☐ the user “golombek” can perceive the event because .....*
- ☐ the user “volker” can perceive the event because ....*
- ☐ the user “golombek” gets an icon displayed, in case .....*
- ☐ the user “golombek” gets a pop-up window displayed, in case .....*

