Tailorability and Usability Engineering: A Roadmap to Convergence

Alex Totter, Chris Stary

Department of Business Information Systems
Communications Engineering
University of Linz
Freistädterstraße 315
A-4040 Linz
Austria
Email: totter@ce.uni-linz.ac.at
URL: http://www.ce.uni-linz.ac.at

ABSTRACT

Tailorability enables user interfaces to be adapted to particular needs of end users and organizations. When developers target towards the construction of user interfaces providing that kind of flexibility, they experience support at the level of testing rather than designing. In search for design support usability evaluation techniques might be used to provide early feedback from end user to designers. In this paper we put methodological and conceptual knowledge from usability evaluation into the context of designing tailorable user interfaces. In doing so, tailorability is related to existing principles of human-centered design. As a consequence, tailorability in its very technical nature has to be considered as an enabling feature for individualization. Thus, it facilitates the implementation of principles of human-centered design. Becoming part of design tools ensures final convergence of tailorability and usability engineering.

Keywords: Tailorability, usability engineering, comparative analysis, conceptual framework, definition.

INTRODUCTION

Tailorability is traditionally described as a feature of interactive software that allows the change of certain aspects of the software in order to meet different user characteristics and requirements. It is widely agreed that the design of tailorable systems is an important future challenge (Stiemerling et al., 1997; Mackay, 1991; Dangelmaier et al., 1999; Malone et al. 1995), since tailorable systems support the structured recognition of the complexity and dynamics of tasks and organizations, as well as of inter- and intra-individual differences between end users.

Tailoring activities can be performed at different system components, requiring different knowledge / understanding of the users carrying out the tailoring activity (Mørch, 1997, Ulich et al., 1991). Sample interactive software systems that support different demands of end-users to tailor user interfaces according to their needs are: OVAL (Malone et al., 1997), the HyperCard programming system for the Macintosh, and spreadsheet applications in general.

However, several authors have pointed out that the design of tailorable interactive systems lack development procedures, frameworks, commonly agreed concepts, and effective support, since designers have to take into account several problems which classical design methodologies do not address (Kahler, 1995; Appelt et al., 1997, Stiemerling et al., 1997; Mørch et al., 1997). In this paper, to overcome these deficiencies, we put methodological and conceptual knowledge from usability evaluation into the context of designing tailorable user interfaces. The roadmap to convergence is enabled through evaluation activities becoming part of design activities. Evaluation activities thus should provide early feedback from end users to designers and foster direct input to developers. Our presented roadmap to convergence does not only contain an analysis of terms and interpretations of tailorability, but also a framework how to analyze existing usability evaluation instruments with respect to structured design support and design tool development.

In this paper we particularly
(i) analyze existing interpretations of tailorability
(ii) derive characteristic properties from the results of (i) in a structured way
(iii) contrast and supplement the results of (ii) with the results of a structured analysis of evaluation techniques addressing usability design principles.

The structure of the paper is as follows: We first review related work in the subsequent section with respect to the steps listed above, linking design activities for tailorable systems with evaluation activities. Steps (i) and (ii) are captured in the section on Capturing the Polymorphism of Tailorability. In the section on The Roadmap To Convergence With Usability Engineering we start out with analyzing existing usability engineering techniques and principles, and their contributions to implement or evaluate tailorability, thus, preparing to implement step (iii). We subsequently compare these results with the elements found in step (ii) and come up with an integration of results. Finally, we discuss the results from (ii) and (iii) with respect to the development of design support tools to construct tailorable software systems.

RELATED WORK

In the following we first review work that has already been performed on analyzing tailorability. This part addresses steps (i) and (ii) of our procedure given in the introduction of the paper. We proceed with related work in the field of analyzing usability evaluation techniques. This part addresses the initial task to be performed in step (iii) given above. Finally, we give evidence that evaluation and design activities have to be intertwined, in order to build tailorable systems.

Towards a Framework for Content Analysis. In search for a structured (re)presentation of understandings and interpretations of tailorability, the most comprehensive study has been performed by Ulich et al. (1991). They classified tailoring activities along two dimensions, identifying (common) characteristic properties of tailorable systems as follows:

Modifications of system components:
- input devices: input devices can be selectable (e.g. mouse or keyboard);
- screen layout: size, positions, colors, etc. of windows, menus, texts etc. can be modified;
- data display mode: display information can be changed qualitatively; information can be shown or hidden, alternative display modes (text/graphic) are selectable;

- scope of commands: commands can be included or excluded from a predefined repertoire, but the definition of new commands is not possible;
- command structure: new commands can be developed (e.g. by means of a macro facility);
- individual applications: new applications with dialog structures can be developed;

Methods for accomplishing user tailorability:
- selectable alternatives: different procedures for accomplishing a goal are implemented into the system in a parallel fashion and can be selected (but not modified) by the user;
- online configuration commands: user tailorability can be accomplished by means of special commands included in the application;
- configuration program: modifications can be accomplished interactively by means of a special program or module outside the application;
- configuration file: modifications can be accomplished by editing a special configuration text file with a text editor.

Although tailorability is mainly defined as a technical feature, the structure tells us something about the WHAT has to be achieved with respect to tailorability – see *modifications* part above, and the HOW the addressed properties can be implemented – see *method* part above. Hence, any comparative analysis of tailorability should distinguish between semantic and pragmatic facets.

A Framework for the Comparative Analysis of Usability Evaluation Techniques. There exist few frameworks to compare usability engineering techniques with respect to several perspectives. Most of the existing schemes try to classify techniques for evaluation in very general terms, such as Wixon et al. (1997), McGrath (1995), traditionally based on empirical terms, or to focus on particular, more or less arbitrarily selected, aspects, lacking structural considerations, such as Lewis et al. (1997), and Oppermann et al. (1997). However, a particular multi-perspective framework has been designed to demonstrate the coverage of the usability evaluation space, and to indicate differences on how to evaluate usability: Wixon et al. (1997, p. 681ff.) introduce several dimensions along which usability evaluation techniques can be classified. The authors point out that usability is a multifaceted concept that depends on users, the environment and task characteristics. These parameters allow to categorize evaluation techniques according to the following dimensions and ranges of values:
- ***Formative versus summative methods***: Formative methods are used to create a design, whereas summative methods are used to evaluate a design.
- ***Discovery versus decision methods***: Discovery methods are aiming at discovering how users work, behave, or think and what problems they have. Decision methods are used to structure interface designs or to choose between dialog elements. This polarization is sometimes called qualitative (i.e. discovery) in contrast to quantitative evaluation (i.e. decision).
- ***Formalized versus informal methods***: Some methods are highly structured and have been described formally.
- ***User involvement versus user exclusion***: Methods differ to the extent to which users are involved in evaluation, analysis and design.
- ***Component evaluation versus complete evaluation***: Some methods cover all the steps required to complete designs with respect to usability.

Unfortunately, this categorization neither provides much information in how far the evaluation techniques provides adequate feedback for the design of tailorable systems, nor are the

addressed dimensions logically independent, as it could be exemplified for the techniques analyzed in the course of this study (see also Table 1). In practice, techniques tend to overlap in certain areas defined by these dimensions. Hence, for the objectives of our study, a novel framework for analysis and structured representation had to be developed.

Bridging the Gap between Evaluation and Design. Few approaches to support the ***design*** of tailorable systems have been introduced, such as participatory and evolutionary design involving the users of an application actively in the design process and giving them the opportunity to articulate their requirements. As part of these techniques early usability testing is introduced to evaluate the usability of a software system (Hackos & Redish, 1998). However, the existing literature does not report on usability evaluation techniques that actively support the design of tailorable systems, e.g. through automatically executing guidelines stemming from a styleguide, as soon as specification starts.

The particular recognition of the lack of (pro)active design support motivated us to overcome deficiencies to implement the ideas of tailorability early in the development process. In order to develop techniques and tools supporting the design of tailorable systems the developed roadmap is essentially based on

(i) a classification and categorization of existing approaches to achieve tailorable systems – the list of properties is given at the end of the following section, and
(ii) a structured representation of existing usability evaluation techniques (referring to tailorability issues).

This procedure allows us to conclude with a set of requirements for proper tool development.

CAPTURING THE POLYMORPHISM OF TAILORABILITY

Scanning the literature, several descriptions of tailorability can be found: For instance, Appelt et al. (1997) state, that "... the interface of a computer system is called tailorable when users can ***adapt*** the interface (and the system behind the interface) to their particular situation and their particular needs". To perform this activity, the tailoring of a user interface is described as some sort of a programming task which requires certain programming skills of the tailoring user, classifying users, according to their programming skills into workers, tinkerers, and programmers (introduced by MacLean et al. 1990).

Mackey (1991) refers to tailorability in terms of ***customization*** software defined as "... having mechanisms that allow users to customize their personal software environment without writing code, with changes that persists across sessions". This kind of tailoring has been identified as being most frequently performed, for example, setting default preferences for user interface layout and configuration options.

Other authors introduce the term ***end user programming***, to describe tailorability as "... system development undertaken by users to further develop an existing system to needs that were not accounted for in the original system" (Mørch, 1997), extended to reach software below the user interface and allow different levels of a system to be tailored through extensions, by writing new program code.

Malone et al. (1995) let end users become designers by giving them end-user programming tools, calling them "radically tailorable systems, allowing end users to create a wide range of

different applications by progressively modifying a working system. They "... use the term 'tailorable' to mean that these systems can be changed without ever 'really programming'. More specifically, by tailorable we mean that end users can progressively modify a working system without ever having to leave the application domain to work in separate underlying 'programming' domain".

Only a few authors relate the technically oriented concept of tailorable user interfaces to software ergonomic principles, such as flexibility, to support user-oriented system design. Kahler (1995) and Ulich et al. (1991) suggested to introduce the tailorability approach ".. to reach the necessary flexibility of information technology" in general.

Briefly summarized, the understanding of tailorability can be characterized in general as to address the modifications/changes of the functionality, look and feel of an interactive software system, capturing different user requirements as well as complex and dynamic work settings (such as changes in task accomplishment and organizational settings). Synonyms have been identified, referring to the concept of user interface tailorability as customization or adaptation.

Moreover, different levels of adjustment for different needs and qualifications are suggested (Kahler, 1995), such as recording of macros in word processors to automate sequentially executed tasks; the implementation of an access policy using mechanisms for discretionary access control or changing the screen to the current user's favorite color.

Taking into account the findings by Ulich et al. (1991), namely considering the structure telling us something about WHAT has to be achieved with respect to tailorability – (modifications part of the study), and the methods telling us something about HOW the addressed properties can be implemented (method part of the study), we are able to come up with a descriptive framework of tailorability. It comprises the following elements: synonyms, objectives, roles, features, and levels of adjustment. In the following we exemplify assignments of findings to each of the elements of the framework:

Synonyms:

- adaptation or
- customization.

Objectives:

- meeting user needs;
- persistence of changes across sessions;
- progressively modifying working systems.

Roles involved:

- primarily end users qualifying them as developers, and
- developers enabling the qualification and instantiation process.

***Features* of the interactive** software system with respect to tailorability

- non-programming means (no code writing) ;
- progressive mechanisms.

Levels of adjustment:

- system components: input/output devices, scope and structure of commands etc.;
- interaction modalities.

THE ROADMAP TO CONVERGENCE WITH USABILITY ENGINEERING

In order to compare the results from the descriptive analysis of tailorability with techniques from usability engineering, we have to analyze existing usability engineering techniques. We will start with in how far their underlying usability evaluation principles refer to tailorability. We proceed with the structured representation and analysis of some of the major usability engineering techniques, providing the roadmap of convergence.

Usability Engineering – Fundamentals

Usability has been referred to as the quality of a product in use and is defined in the ISO 9241-11 standard (ISO, 1997) as follows:

- Usability of a product is the extent to which the product can be used by ***specified users*** to achieve ***specified goals*** with effectiveness, efficiency, and satisfaction in a ***specified context of use***.

Karat (1997) elaborates upon the definition given above:

- The usability of a product is not an attribute of the product alone. It is rather an attribute of interaction with a product in a context of use.
- A usability evaluation technique is a process for producing a measurement of usability.

These explanations identify usability as a complex, multidimensional concept, requiring the integrative consideration of cognitive components (specified users), the organization of the environment (specified as a set of goals in a specified context of use, such as work), technical features (the product), and their intertwining (interaction). In order to capture these requirements, more concrete usability evaluation principles have been developed as a basis for implementing the measurement of usability. They have been incorporated in usability engineering techniques. Hence, we will analyze usability evaluation techniques focusing on two questions:

(i) Which of their underlying usability principles can be correlated to the derived characteristics of user tailorable systems (see previous section) to provide adequate input for the design of tailorable systems?
(ii) In how far is the structure of the evaluation technique (roles, activities etc.) appropriate to generate inputs for system design?

Usability Engineering – Techniques for Evaluation

The selection of the techniques has been performed according to their objective, namely measuring usability, and their common availability. After a brief description of each of the selected techniques, a widely used classification framework is applied to categorize the techniques' coverage of the usability evaluation space, thus justifying their selection. However, as already discussed in the section Related Work, this framework does not meet the objectives of this study. Hence, we will continue with the descriptive framework introduced in the previous section to complete the study.

The selected techniques for usability evaluation are:

- ***Cognitive Walkthrough*** (Lewis & Wharton, 1997): This usability inspection method focuses on ***evaluating a design for ease of learning***. It attempts to provide a detailed, step by step evaluation of the user's interaction with an interface in the process of carrying out a

specific task. The process of the cognitive walkthrough includes a preparation phase, the analysis and the follow up phase. In particular, the analysts ask the following four questions:

- Will the user try to achieve the right effect?
- Will the user notice that the correct action is available?
- Will the user associate the correct action with the effect that user is trying to achieve?
- If the correct action is performed, will the user see that progress is being made towards solution of the task?

The technique seeks to identify mismatches between users' and designers' conceptualization of a task, poor choices of wording for menu titles and button labels, and inadequate feedback about the consequences of an action. The method is performed by analysts and reflects the analysts' judgments. They identify problems by tracing the likely mental processes of a hypothetical user.

- ***EU-Con*** (Stary et al., 1998): The EU-CON technique has been developed to implement the ***EU-directive 90/270/EEC*** (EU, 1990) on man-machine communication. According to the directive the software has to meet several minimal requirements, namely:
  - task conformance,
  - ease of use,
  - ***adaptability towards human capabilities, skills and experiences,*** and
  - support of human-information processing.

  Adaptability is understood as the provision of mechanism to enable developers and end users to react on dynamically occurring requirements. These requirements might concern tasks, user characteristics, interaction modalities or a combination of those items. In addition, an interactive software system is considered to be adaptable to individual needs (based on the principle of individualization) in case it can reflect user needs and skills with respect to a certain task.

  EU-Con follows a two step strategy. First, the users are guided through a measurement procedure, in order to indicate usability problems. Then, evaluators identify the reasons for the problems users are experiencing, and try to remove barriers to effective and efficient interaction. The procedure consists of four phases: preparations, execution, tuning, and rework. Within each phase several steps have to be performed. The evaluation process is supported be a questionnaire to be filled in by users, a guide for evaluation and a handbook for evaluation and engineering.

- ***ISO 9241 evaluator*** (Oppermann et al., 1997): This technique is an example of a expert-based evaluation method for ***conformance testing with the ISO 9241 standard part 10-17***. The criteria of the ISO 9241 standard part 10-17 are:
  - suitability for the task;
  - self-descriptiveness;
  - controllability;
  - conformity with user expectations;
  - error tolerance;
  - ***suitability for individualization***: "Dialogue systems are said to support suitability for individualization if the system is constructed to allow for adaptation to the

users' needs and skills for a given task....The overall objective should be to provide mechanisms which allow the system to be adapted to the individual
    - knowledge and experience of the computer
    - knowledge and experience of task domain
    - language and culture
    - perceptual / motor abilities
    - cognitive abilities of the user"
- suitability for learning.

The ISO 9241 evaluator is a guideline-oriented evaluation technique that tests the multiparty standard ISO 9241 in about 300 items. The test items are structured in a two dimensional space defined by technical components and software ergonomic principles. Each item checks a particular aspect of ergonomic requirements specific for the given component and principles. The primary scope of the evaluation with the ISO 9241 evaluator is the user interface of a software system. The ISO 9241 evaluator offers support for testing, for documenting the testing, the evaluation, and for reporting the results. To collect information about the context of use, a simplified workplace analysis and a questionnaire exploring user characteristics have to be administered.

- ***Heuristic Evaluation*** (Nielsen, 1994): This technique is a usability engineering method for locating usability problems in user interface design in a way that it can be performed as part of an iterative design process. Heuristic evaluation involves the participation of a small group of usability experts, who ***examine the interface and judge its compliance with recognized usability principles***, so called usability heuristics:
    - visibility of the system status;
    - match between system design and the real world;
    - user control and freedom;
    - consistency and standards;
    - error prevention;
    - recognition rather than recall;
    - ***flexibility and efficiency of use:*** Accelerators - unseen by novice users - may often speed up the interaction for the expert user to such an extent that the system can cater both inexperienced and experienced users; Allow users to tailor frequent actions.
    - aesthetic and minimalist design;
    - help users recognize, diagnose, and recover from errors;
    - help and documentation.

  In the course of an evaluation session the evaluator studies the user interface several times, inspects the various dialog elements, and checks their structure and behavior against a list of recognized usability principles. Evaluators are supplied with a typical scenario of use, listing the various steps a user would take to perform a sample set of realistic tasks. Such a scenario should be constructed on the basis of a task analysis performed with actual users under actual work conditions, in order to generate utmost representative scenarios. The result is a list of usability problems in the interface, with reference to those usability principles that were violated by the design.

Having briefly outlined the techniques under investigation, we will now cluster them in Table 1 according to Wixon et al. (1997, p. 681ff.) to justify their selection. As it can be seen the

selected techniques do not only address tailorability (level of principles), but also capture a large variety of known methods for evaluation (level of method).

Table 1: Categorizing the Techniques under Investigation

Method **Classification**	**Cognitive walkthrough**	**EU-Con**	**ISO 9241 evaluator**	**Heuristic Evaluation**
Formative **Summative**	Formative	Formative Summative	Summative	Summative
Discovery **Decision**	Discovery	Discovery Decision	Decision	Discovery Decision
Formalized **Informal**	Formalized	Formalized	Formalized	Informal
User involvement **Expert-based**	Experts	User involvement	Experts	Experts
Complete **Component**	Component	Component	Component	Complete

The Roadmap – Structured Representation

To enable the analysis of usability evaluation techniques according to the objectives of the study a structured scheme of representation has been developed. The scheme comprises (i) the activities to be performed in a certain sequence, (ii) the tools that have been developed to support the particular evaluation technique, and (iii) the persons (roles) that perform the evaluation. Based on this information, a comparison with the results from the content analysis can be performed.

Cognitive Walkthrough: The purpose of this technique is to suggest to the designer where the design is likely to fail and why. It examines a design for ease of learning. The analysts think about the mental processes of users. Since end users are not involved, the validity of the analysts' assumptions about the mental models of the end users are not compared to the end users' mental models.

The focus on the tailorability is not given, since the designer is responsible for determining ***one*** correct sequence of actions, no matter whether there actually exists more than one (best) way of performing the task. Furthermore, in addition to the four questions, the technique does not offer tools that support the analysts in how to perform the evaluation.

EU-Con: Since EU-Con has been developed according to the directive, the usability evaluation criteria ***"adaptability towards human capabilities, skills and experiences"*** has been turned into operational definitions for measurement, meeting the characteristics of tailorable systems. Through the participation of end users, and additional questions collecting information about end users, the identification of inter-individual differences is supported.

Moreover, direct feedback for the designer of a tailorable system is provided through the discussion between end users and designers on how to improve the interface, as well as the support through the handbook for engineering (which provides suggestions for improving the user interface to compensate for specific categories of problems).

Table 2: Structured representation of the Cognitive Walkthrough

Roles	Activities	Tools to support the technique
	Define inputs to the walkthrough	
Analysts	• define assumed user background	• ?
Analysts	• choose sample tasks	• task should be important, task should be realistic
Designer	• specify correct action sequences tasks	• rough guidelines, most common way
Designer	• determine interface states along the sequences	• stepping through the interface
?	**Convene the analysis**	
	Walk through the action sequences for each task	
Analysts	• for each correct action construct a success story that explains why users would choose that action	• ask four questions
	• use a failure story to indicate why a user would not choose that action	• ask four questions
	Record critical information	
Analysts	• make assumption about the user population	
	• note side issues and design changes	
	• tell credible success stories	
	• tell failure stories (record problems, reasons)	
	• consider and record design alternatives	
Designer	**Modify the interface design to eliminate problems**	

Table 3: Structured representation of the EU-Con

Roles	Activities	Tools to support the technique
	Preparation phase	
supervising evaluator	• brief the user	• briefing sheet
	• handle out questionnaires for evaluation	• questionnaire
end user	• identify task	• manual that guides the user through the entire identification and questioning-answer process
end user	**Execution phase**	
	• complete the questionnaire	• questionnaire
	Tuning phase	
supervising evaluator	• utilize the questionnaires	• alert sheets
	• interpret the results	• handbook for evaluation
		• handbook for engineering
	Rework phase	
supervising evaluator	• identify possible reasons for usability problems	• handbook for evaluation
		• handbook for engineering
supervising evaluator end user designer	• discussion problems in detail with end users	
	• implement improvements at the user interface	

ISO 9241 evaluator: This expert- and guideline-oriented technique evaluates the user interface through conformance testing with the ISO 9241 standard part 10-17. Since one of the principles focuses on the ***suitability for individualization,*** tailoring aspects are considered. In

particular, through the two dimensional space, defined by technical components and software ergonomic principles, a direct relation to the modifications of system components to be tailored can be given. Unfortunately, no end-users are involved. Therefore, the suggestions made by experts concerning tailorability functions can only be given at a very general level.

Table 4: Structured representation of the ISO 9241 evaluator

Roles	Activities	Tools supporting the technique
evaluator (expert with good expertise about human factors)	**Preparation phase** • gather typical user characteristics • gather typical task characteristics	• questionnaire • simplified work place analysis
?	• chose actual software / user interface	?
evaluator	• specify test situations for each item	• view editor
evaluator	• record test situations for each item	• ISO 9241 evaluator
evaluator	• evaluate each test item for the test situations	• ISO 9241 evaluator • EVADIS evaluation guide
evaluator	• write an explanation for the evaluation	• integrated capture tool
evaluator	• capture detected deficiencies	• integrated capture tool
evaluator	• write evaluation report	• text editor

Heuristic Evaluation: This task- and user-independent technique directs the attention mainly to the characteristics of the interface. Since one of the heuristics focus on ***"flexibility and efficiency of use***" tailoring aspects are considered. Unfortunately, through the lacking participation of end users, only general assumptions can be made. Although in the debriefing session the outcome of the evaluation is discussed and suggestions for improving the interface can be gathered, no tools have been developed to support the aforementioned processes or any re-design (see table 5).

Table 5: Structured representation of the heuristic evaluation

Roles	Activities	Tools to support the technique
Heuristic evaluation expert designer?	**Pre-Evaluation training session** • briefing on the method • briefing on the domain • briefing on the scenario	? ? • task analysis
evaluator (usability and or interface specialists) evaluator	**Actual evaluation** • find as many usability problems in the interface as possible • state what established usability principle was violated by each usability problem	• scenarios • usability heuristics
evaluator	**Debriefing session to discuss outcome of the evaluation** • modify the heuristic evaluation method to include advice for the redesign phase	?
evaluator	**Severity rating phase** • assess the severity of the usability problems that had been found in the evaluation session.	• rating scale

In a brief summary we can conclude that many techniques

(i) can be related to issues of tailorability, either through their direct relationship to standards or generally acknowledged principles, or through their freedom to select particular principles, as experts feel appropriate, for the purpose of evaluation;

(ii) are incomplete, either with respect to role definitions, activity specifications, and/or tool support;

(iii) do not address the incorporation of the results of evaluation into other phases or steps of development than usability testing, although some of the authors claim to address design issues;

(iv) tend to abstract from tasks and needs from end users. They rather assume that deficiencies at the user interface can be detected by independent experts and/or task- and user-independent settings for evaluation.

In general, it has to be recognized that usability engineering is understood mainly in terms of usability testing. This fact might explain the lack of explicit and well grounded relationships to design activities.

Striving for Convergence

Based on the results from the descriptive content analysis as well as those from structuring existing evaluation techniques the following commonalties and required extensions of usability techniques become evident:

Roles involved:

- tailorability activities should be performed by end users requiring different levels of programming skills;
- some of the existing evaluation techniques do not involve end users;

From these results we can conclude:

- End users as well as developers might be part of the usability engineering process.
- However, developers have to change their roles toward enablers of qualification and functions, based on initial, direct user inputs.

Objectives

- tailorability activities should capture different user requirements as well as complex and dynamic work settings;
- usability evaluation techniques mainly measure usability based on usability evaluation principles, such as ease of learning, ISO standard 9241 part 10 ...;

From these results we can conclude:

- Some of the usability evaluation techniques measure tailorability only implicitly through very general usability principles, not focusing on different user characteristics of taking into account dynamic work settings.

Features

- usability evaluation techniques hardly measure interactive systems at the features level.

From these results we can conclude:

- Features of the interactive software system with respect to tailorability, such as non-programming means (no code writing) or progressive mechanisms can become part of evaluation, in case concrete activities, as exemplified in Ulich et al. (1991) through commands and files, become available.

Levels of adjustment

- usability evaluation techniques evaluate interaction devices and modalities at a very general level, frameworks for systematic investigations of levels of adjustments are missing.

From these results we can conclude:

- Levels of adjustment, such as proposed by Ulich et al. (1991), are currently restricted interaction devices and modalities. They have to be enhanced with the conceptually addressed ones, namely scope and structure issues.

For the development of proper design tools the following list of requirements can thus be derived:

(i) Several roles have to be provided. They range from end users to evaluators – see the structured representation of evaluation techniques.
(ii) The principle of tailorability with respect to task structure and accomplishment, user capabilities, needs, and preferences, and with respect to interaction modalities have to become part of the tool, e.g. through a hyper-linked knowledge base.
(iii) The target user population of such tools is not only the development team, but rather the end user who should become qualified to use tailorability functions.
(iv) The tool has to provide process support to achieve tailorability.

CONCLUSIONS

Becoming aware of the demand for tailorable software systems, we set out exploring the understanding of tailorability and corresponding support for design and evaluation. Our first impression has turned out to be the regular situation in software development: Although standardization bodies and usability evaluators care about features of tailorability, there is a lack of operational support for designers, who have to be considered to be the ultimate pre-processors to programmers. In case programming is not preceded by clear tailorability objectives and accurate design specifications developers have to wait until usability testing for user feedback to measure the success of their efforts, rather than being able to check their code against tailorability specifications and corresponding operational requirements.

Since developers find themselves lost with respect to tailorability design support we have developed a framework and roadmap to investigate how to take into account existing interpretations of tailorability and respective techniques, in order to develop specifications for tailorability design support tools. We have also been driving some of the roads on the map, in order to proof our methodological concept. We have also paved some of the roads, giving some results.

Although we were able to find some explicit correspondences between the two fields involved, some links might have remained implicit ones. They have to be elaborated in further studies of this kind. In particular, a more extensive literature survey has to be performed, and a larger amount of usability engineering approaches has to be represented and compared in a structured way. Moreover, concepts related to tailorability via the organization of work, the social setting of end users, and cognitive capabilities, have to be analyzed and put into the context at hand.

However, some results of the process we started can be given by that time:

- Tailorability is not merely a set of technical features to be adjusted. It requires a process, different levels of adjustment and has to take into account the entire 'reality' of end users, as perceived by them.
- Tailorable systems require design techniques and tools, with strong emphasis on contextual knowledge and the evaluation of design results.
- The design of tailorable systems requires a closer look at and involvement of the actual end users. Hence, the development process **as well as** the result require end user involvement and participation in a fully competent and qualified way.

As a consequence, the development process might not occur in a structured and straightforward way as developers might wish to experience when designing tailorable systems. A shift of roles is required. Developers have to become enablers to provide end users with capabilities for tailoring **and** qualify them using these functions, since tailorability has to be understood as end user empowerment from the very beginning of development instead of adding 'just-another' set of technical functions as part of the user interface or the application's functionality.

References

Appelt, W.; Hinrichs, E. and Woetzel. G. (1997): Effectiveness and Efficiency: the Need for tailorable User Interfaces on the Web. In: Proceedings of the Seventh International World Wide Web Conference, Brisbane, Australia, 14th-18th August, 1997.

Dangelmaier, W.; Hamoudia, H.; Klahold, R. F. (1999): Domain Preferences for End-User Tailoring in Shared Workflow Interfaces. Workshop Implementing Tailorability in Groupware, in conjunction with the International Joint Conference on Work Activities Coordination and Collaboration WACC'99, San Francisco, CA, USA, February 1999.

EU (1990): Directive 90/270/EWG on Man-Machine Communication, in: Directive 29.5.1990 about Safety and Health of VDU-Work, Official EC-Newsletter, 33, L156, Minimal Standards, Par. 3, p. 18, 21.6.

Kahler, H. (1995): From Taylorism to Tailorability. Supporting Organizations with tailorable Software and Object orientation. In: Anzai, Y.; Ogawa, K.; Mori, H.: Symbiosis of Human and Artifact. Human and Social Aspects of Human-Computer Interaction. Proceedings of the Sixth International Conference on Human-Computer Interaction, (HCI International'95), Tokyo, Japan, 9-14 July 1995, Volume 2, pp. 995-1000, Elsevier: Amsterdam.

Karat, J. (1997): User-Centered Software Evaluation Methodologies, in: Helander M.; Landauer, T.K.; Prabhu, P. (eds.): Handbook of Human-Computer Interaction (2nd ed.); Elsevier Science B.V.; pp. 689-704.

Lewis, C.; Wharton, C. (19997): Cognitive Walkthroughs, in: Helander M.; Landauer, T.K.; Prabhu, P. (eds.): Handbook of Human-Computer Interaction (2nd ed.); Elsevier Science B.V.; pp. 717-732.

McGrath, J.E. (1995): Methodology Matters: Doing Research in the Behavioral and Social Science, in: Baecker, R.M.; Grudin, J.; Buxton, W.; Greenberg, S. (eds.): Readings in Human-

Computer Interaction: Towards the Year 2000 (2nd ed.), Morgan Kaufmann Pub.: San Francisco, pp. 152-169.

McLean, A.; Carter, K.; Lövstrand, L., Moran, T. (1990): User-tailorable Systems: Pressing the Issue with the Button, in: Proceedings of the Conference of Computer-Human Interaction, CHI'90, April, 1-5, 1990. Seattle, Washington, ACM Press, New York, 1990, pp. 175-182.

Mackay, W.E. (1991): Triggers and Barriers to Customizing Software. In: Proceedings of ACM CHI '91 Human Factors in Computing Systems. New Orleans, Louisiana: ACM/SIGCHI, pp. 153-160.

Malone, T.W.; Lai, K.; Fry, C. (1995): Experiments with Oval: A Radically Tailorable Tool for Cooperative Work. In: ACM Transactions on Information Systems, Vol. 13, No 2, April 1995, pp. 177-205.

Mørch, A. (1997): Aspects of End User Tailoring: In: Workshop on Tailorable Groupware: Issues, Methods, and Architectures. GROUP'97, International Conference on Supporting Group Work: The Integration Challenge November 16-19, 1997, Embassy Suites Hotel, Phoenix, Arizona, USA.

Nielsen, J. (1994): Heuristic Evaluation, in: Nielsen, J.; Mack, R.L. (eds.): Usability Inspection Methods, John Wiley & Sons, Inc.: New York, pp. 25-62.

Opperman, R.; Reiterer, H. (1997): Software Evaluation using the 9241 Evaluator, in: Behaviour & Information Technology, 1997, Vol. 14, No. 4/5, pp. 232-245.

Stary, Ch.; Riesenecker-Caba, T.; Flecker, J. (1998): Implementing the Directive for VDU work - the EU-state of the art, in: Behaviour & Information Technology, 1998, Vol. 17, No. 2, pp. 65-81.

Stiemerling, O.; Kahler, H.; Wulf, V. (1997): How to make Software Softer - Designing Tailorable Applications. In: Proceedings of DIS'97: Designing Interactive Systems: Processes, Practices, Methods, & Techniques, 18-20 August 1997 Grand Hotel Krasnapolsky Amsterdam, The Netherlands 1997, pp. 365-376.

Ulich, E.; Rauterberg, M.; Moll, T.; Greutmann, T.; Strohm, O. (1991): Task Orientation and User-Oriented Dialog Design. In: International Journal of Human-Computer Interaction 3(2), pp. 117-144.

Wixon, D.; Wilson, Ch. (1997): The Usability Engineering Framework for Product Design and Evaluation, in: Helander M.; Landauer, T.K.; Prabhu, P. (eds.): Handbook of Human-Computer Interaction (2nd ed.); Elsevier Science B.V.; pp. 653-688.