Telematics Applications Project IE 2016

Information Engineering Usability Support Centres

WP 6

Deliverable D6.2

Handbook of User-Centred Design

Jurek Kirakowski, HFRG and Nigel Bevan, NPL (Eds)

Final Version

February, 1998

Summary

This handbook provides an introduction to the key principles and activities of user-centred design as advocated by the ISO 13407 standard. The handbook also contains descriptions of a range of established methods which can be used to evaluate the usability of a product or system in order to achieve quality in use. The description of each individual method includes details on the pre-requisite knowledge required as well as an indication of the equipment and resources that will be needed to put them into practice. Advice is provided to guide the selection of appropriate methods and an overview of relevant standards and guidelines is also included. The handbook is intended for those responsible for commissioning or carrying out usability work during the development of interactive systems.

Keywords: usability, methods, quality in use.

Acknowledgements

This handbook has benefited from the important contributions made by members of the INUSE and RESPECT projects, and particularly the efforts of:

Albert Arnold	WITLab, Delft University of Technology, The Netherlands
Stefana Broadbent	CB&J, France
Nigel Claridge	Nomos Management AB, Sweden
Ian Curson, Owen Daly-Jones and Cathy Thomas	NPL Usbility Services, UK
Jonathan Earthy	Lloyd's Register of Shipping, UK
Jan Heim and Tor Endestad	SINTEF Unimed Rehab, Norway
Keith Hurley and Nicola Barry	Human Factors Research Group, University College Cork, Ireland
Martin Maguire	HUSAT Research Institute, Loughborough University, UK
Paulus Vossen	Fraunhofer Institute, Germany

Executive Summary

This handbook on user-centred design is intended for those responsible for commissioning or carrying out usability work during the development of interactive systems. It consists of 5 chapters which are summarised below.

Chapter 1: A user-centred approach to design and assessment

The first chapter presents a general introduction to the design approach advocated in the emerging ISO 13407 standard - Human Centred Design Processes for Interactive Systems. Special emphasis is placed on the need to develop software which is usable, i.e. effective, efficient and satisfying to use. It is suggested that the way to achieve these qualities is to adopt a user-centred approach to system design. Where appropriate the relationship between these user-centred design activities and the various types of method discussed later in the handbook is indicated.

Chapter 2: Introduction to usability methods

This chapter presents an overview of the range of user-centred design and assessment methods discussed in chapter 3. It is important to note that most of the methods included in the handbook are applicable to a wide range of systems and as such can be used to obtain design feedback, metrics and subjective data regardless of the particular application domain. Many of the methods can also be used to study different user groups, such as novices and experts. The methods selected for inclusion represent established approaches which have been successfully applied within industry. A short list of other methods of potential value is also included in this chapter. The term 'method' is used loosely to refer to techniques and processes as well as rigorously specified methods.

Chapter 3: Individual method descriptions

This chapter describes twenty established usability methods. The description of each individual method includes details on the key steps to be taken as well as an indication of the resources that will be needed to put them into practice. The methods are organised into 6 categories: planning, guidance and standards, early prototyping, expert-based evaluation, user performance evaluation, and subjective assessment.

Chapter 4: Selecting an appropriate method

This chapter provides guidance to aid in the selection of appropriate methods, based on the costs and benefits of using a particular method. Relevant issues include the kinds of information to be gained by adopting a particular approach, the resources required and the inherent benefits and limitations of different approaches. Many of these criteria are presented in a summary table and a process for using this summary matrix to choose usability methods is given. As a further source of guidance a number of scenarios for usability activity are included.

Chapter 5: Standards and guidelines in user-centred design

This part of the handbook summarises the content of a range of international standards related to user-centred design. Other sources of guidance such as styleguides are also discussed.

Contents

1. A user-centred approach to design and assessment

1.1 Overview

This chapter presents design principles for those who are responsible for developing software solutions. It places a special emphasis on the need to develop software which is usable, i.e. effective, efficient and satisfying to use. It is suggested that the way to achieve these qualities is to adopt a user-centred approach to software design and development. This approach normally involves a number of key activities throughout the development of the software including involving users, obtaining their feedback on the design and use of the system, and providing prototypes for users to try out and re-designing in light of user feedback and comments. The benefits of this approach include increased productivity, enhanced quality of work, reductions in support and training costs and improved user health and safety.

This chapter presents a general overview of the approach advocated in the emerging ISO 13407 standard - Human Centred Design Processes for Interactive Systems. The reader is first introduced to the basic concepts of user-centred software development, while the key activities which should be undertaken to ensure good design are discussed in the latter part of the chapter. Where appropriate the relationship between these user-centred design activities and the various types of method discussed later in the handbook is indicated.

1.2 What is user-centred design?

As an approach to software development user-centred design focuses specifically on making products usable. The usability of a product is defined in ISO 9241, part 11 (the forthcoming standard giving guidance on usability) as:

> ‘the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use’.

This definition relates to the quality of the interaction between the person who uses the product to achieve actual work and the product or software application itself. The important features of this interaction are *effectiveness*

- how well the user accomplishes the goals they set out to achieve using the system; *efficiency* - the resources consumed in order to achieve their goals; and *satisfaction* - how the user feels about their use of the system.

Developing software with the quality of this interaction in mind is the primary goal of user-centred design. The approach typically entails involving users in the design and testing of the system so that their feedback can be obtained. Prototypes are usually employed to do this and designs are modified in light of the user feedback. Following this process to develop software can result in a number of significant advantages for the software developer, by producing software which:

- is easier to understand and use, thus reducing training costs
- improves the quality of life of users by reducing stress and improving satisfaction
- significantly improves the productivity and operational efficiency of individual users and consequently the organisation.

Initially it may seem that the user-centred approach complicates the software development task, due to the need to make iterative refinements to the software in light of user feedback. However the benefits to be gained are considerable. As well as the advantages listed above the process promotes communication between users, managers and those developing the software and identifies problematic issues early on in the development schedule when it is much cheaper to implement changes. For these reasons alone it is worth adopting a user-centred perspective. In the following section the main principles involved in the approach are examined in greater detail.

1.3 Main principles of user-centred design

Within the field of software development there are numerous methods for designing software applications. While these vary in nature one common trait concerns the stress placed on meeting the technical and functional requirements for the software. This is of course important, however, it is equally important to consider usability requirements if the benefits outlined above are to be realised. The principles described in this section are aimed at incorporating the user’s perspective into the software development process. They are presented as a compliment to existing strategies rather than a replacement. In line with the ISO 13407 standard the key aspects are:

- an appropriate allocation of function between user and system

Determining which aspects of a job or task should be handled by people and which can be handled by software and hardware is of critical importance. This division of labour should be based on an appreciation of human capabilities, and their limitations, as well as a thorough grasp of the particular demands of the task. Naturally this allocation benefits from the input of end-users or their representatives which will also help to ensure that the results are acceptable to the people who will be affected.

- the active involvement of users

One of the key strengths of user-centred design is the active involvement of end-users. The extent of this involvement depends on the precise nature of the design activities but generally speaking the strategy is to utilise people who have real insight into the context in which an application will be used. Involving end-users can also enhance the acceptance and commitment to the new software as staff come to feel that the system is being designed in consultation with them rather than being imposed on them.

- iteration of design solutions

Iterative software design entails the feedback of end-users following their use of early design solutions. These may range from simple paper mock-ups of screen layouts to prototypes with greater fidelity which run on computers. The users attempt to accomplish 'real world' tasks using the prototype and the feedback from the exercise is used to develop the design further.

- multi-disciplinary design teams.

User-centred software development is a collaborative process which benefits from the active involvement of various parties, each of whom have insights and expertise to share. Given the potential contributions to be made by each perspective it is important that the development team is made up from representatives of all those groups who have a 'stake' in the proposed software. Depending upon the circumstances this team may include managers, usability specialists, training and support staff, software engineers, quality assurance representatives and of course the end user themselves, i.e. the people who will use the final product.

1.4 Preparing for user-centred system development

User-centred design can be incorporated into existing software development strategies. It may be reflected in an organisation's procedures or in their quality plans for software development. Naturally the user-centred activities should be planned and managed with the same rigour as other approaches. Plans should identify the scope of the required activities as well as their relationship to any conventional development work being undertaken. The resources in terms of time, equipment, skills and number of participants should also be clearly identified in advance. The method descriptions presented later in this handbook will help to provide this information. Finally it is vital that a user-centred design programme is planned so that sufficient time exists for the user feedback to be incorporated into the development schedule.

In the next section the key activities to be carried out under this approach are discussed.

1.5 Key user-centred design activities

According to the ISO 13407 standard there are four essential user-centred design activities which should be undertaken in order to incorporate usability requirements into the software development process. These are:

- understand and specify the context of use
- specify the user and organisational requirements
- produce designs and prototypes
- carry out user-based assessment.

The activities are carried out in an iterative fashion as depicted in Figure 1, with the cycle being repeated until the particular usability objectives have been attained. The rest of this chapter is devoted to describing the nature of these activities in more detail.

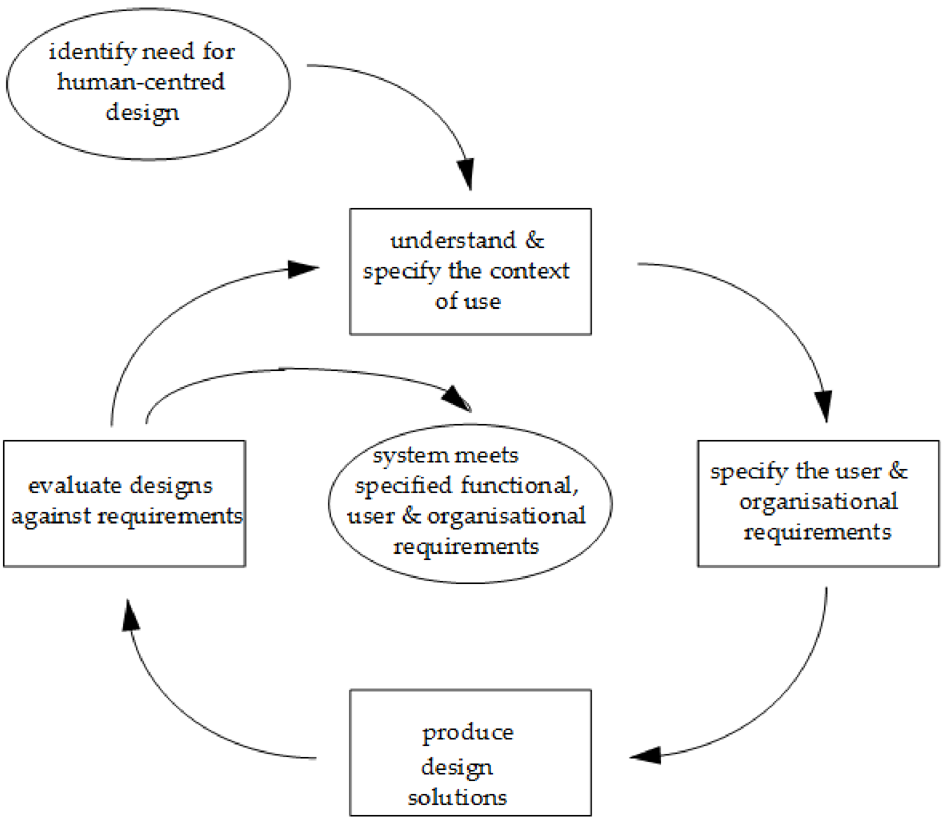

Figure 1. Key user-centred design activities

1.5.1 Understand and specify the context of use for the system

The quality of use of a system, including usability and user health and safety, depends very much upon the context in which a system will be used. For example in an office there are many features of the environment as well as characteristics of the users and tasks which can impinge on the usability of a new software product. Capturing this information is not only important for informing design decisions but also to provide a sound basis for later evaluation activities.

In some cases contextual information may already be known - particularly where a system has been used for some time and is only to receive an upgrade or enhancement of some kind. On the other hand where a new product or system is to be introduced then it will be necessary to collect the relevant contextual information. It is important that as a result of this activity the following aspects are understood:

- the characteristics of the intended users

  Software should be designed with reference to the characteristics of the intended users. Relevant factors may include the knowledge, skill, experience, education, training, physical attributes, habits and motor-sensory capabilities of the people concerned. Where distinctive groups of users exist (such as novices and experts) then their respective attributes should be identified and taken into account.

- the tasks the users will perform

  The new software will be used to achieve certain task goals by the users. It is important that the nature of the tasks and the role to be played by the proposed system are well understood. This information should focus on high-level characteristics such as the goals, frequency and duration of individual tasks rather than on the functional capabilities of the software.

- the environment in which the users will use the system.

  It is also necessary to describe the environment in which the system will be used. This relates to organisational, technical and physical factors. Examples of relevant organisational aspects include the structure of operational teams and the level of autonomy that individual staff experience. Elements of the technical environment such as the hardware and networking which the new software depends upon should also be identified. Finally the physical conditions in which the software will be used must be considered - this can include ambient qualities (e.g. temperature and humidity) as well as potential health and safety issues.

This information can be gathered via a variety of means such as interviews and task analysis. One particularly effective approach is to hold a context meeting in which all those who have insights and experience to share are present. The meeting is held with representatives of all the major stakeholders (e.g. managers and end-users) so that the contextual details can be identified and agreed upon.

The results of this initial activity are embodied in a document which describes the context of use for the proposed software. This details the characteristics of the users, their tasks and their environment. It should also highlight those aspects which could be important for the design of the software. Naturally the content of this document may need to be updated to reflect changes in the context over the course of the development schedule. The purpose of this document is to inform the decisions of the design team, and it should also be used when planning evaluation activities. For further details on how to collect contextual information see the method entry for Usability Context Analysis in chapter 3 of this handbook.

1.5.2 Specify the user and organisational requirements for the system

Requirements elicitation and analysis is widely accepted to be the most crucial part of software development. Indeed, the success of a software development programme can largely depend on how well this activity is done. Alongside the technical and functional requirements for software the importance of usability requirements has been stressed. Building on the context of use description obtained as a result of the previous activity an explicit statement of the user-centred requirements for the new software should be formulated.

The planning methods as well as the guidance and standards methods described in chapter 3 can help define these requirements. In addition the RESPECT project has produced a handbook which offers a framework for capturing user-requirements, details of which are provided in the bibliography presented in chapter 6. Detailed guidance on specifying user and organisational requirements and objectives is also provided in ISO 9241- part 11, but generally speaking the following elements (as listed in ISO 13407) should be covered in the specification:

- identification of the range of relevant users and other personnel in the design
- provision of a clear statement of design goals
- an indication of appropriate priorities for the different requirements
- provision of measurable benchmarks against which the emerging design can be tested
- evidence of acceptance of the requirements by the stakeholders or their representatives
- acknowledgement of any statutory or legislative requirements, for example for health and safety.

It is also important to be aware that the requirements are likely to change over time!

1.5.3 Produce design solutions

Having identified the relevant contextual information and usability requirements the next stage is to explore design solutions by creating simple mock-ups of the proposed system and then later presenting them to a representative sample of users (see section 1.5.4). In the first instance the design solution can be based on previous experience or the advice given in standards and styleguides but it is important to realise that the initial design will be refined in light of user feedback. With this in mind the key goal of this user-centred activity is to:

- simulate the design solution(s) using paper or computer-based mock-ups

Prototyping can be carried out from the earliest stages of the design sequence as well as during later stages. The prototype of a system can be as simple as using basic materials such as paper, card and pencils to create static representations of screens, menus and windows. Alternatively the prototype may run on a computer system and offer the opportunity to interact with a simulation which is quite close to the final product in functionality. However, while realism can have its advantages it is better to avoid making too much of a commitment to a particular implementation.

There are numerous benefits to be gained from using prototypes. The activity fosters greater communication between the development team and end-users. Simple prototypes also allow different design options to be explored prior to coding, whilst allowing future problems to be identified early on in the development process when changes can be implemented more quickly and cheaply. These qualities mean that user requirements can be more readily incorporated into the eventual design.

The methods described in the guidance and standards and early prototyping sections of chapter 3 can be used to support the production of designs and prototypes. Expert-based evaluation methods can also be used to validate early designs and ensure that the later user-based trials do not simply result in the identification of known usability defects.

1.5.4 Evaluate designs against requirements

Once a prototype or a working version of the software is available a user-based assessment of the product can be carried out. This is a very important development activity which can confirm the extent to which user and organisational objectives have been met as well as provide further information for refining the design. As with the other user-centred activities it is advisable to carry out evaluations from the earliest opportunity, before making changes becomes prohibitively expensive. There are a number of discrete steps to this activity which progress from planning the evaluation, collecting and analysing the results and reporting recommendations for change. Iteration of the activity is also advised until design and usability objectives are met. The activity is comprised of the following steps:

- evaluation plan

It is crucial that any evaluation activity is planned for in advance as there are a number of issues which must be taken into account. The first is the identification of relevant stakeholders. It is important to secure their support for the evaluation process and their agreement over when and how the user and organisational requirements will be assessed.

Then, based on the earlier context of use analysis, a group of representative end-users should be contacted with a view to asking them to use the new system to achieve assigned but realistic tasks. The actual number of people required will depend on the particular usability method to be applied. These differ in terms of the rigour of the method and the data that is provided. For the evaluation of simple paper-based prototypes a small group of users (3-5) will be sufficient to gain useful design feedback. Later in design where objective metrics of effectiveness and efficiency may be required then 8-10 participants should be used. In addition to user-based evaluations a quick and economical evaluation of a system can be achieved by having usability experts examine the software. However, while this method is valuable it is not a replacement for having actual users test a system out. The choice of method will also be influenced by time and resource constraints.

At a more specific level the evaluation plan should identify: the roles for those people who will run the evaluation sessions, the aspects of the system to be evaluated, the tasks to be assigned to the users, and the methodology and resources required for the evaluation. It will also be necessary to define the types of information to be gathered and how this will be measured given the interests and needs of the procurers of the software. For more guidance on specifying usability criteria the interested reader is referred to the ISO 9241-11 standard. Finally the planned schedule should also allow sufficient time for processing the results and feeding back any recommendations for change. The usability planning method presented in chapter 3 can be used to support the creation of an evaluation plan.

- data collection and analysis

Once all the resources are available, the tasks have been created, and the users recruited the evaluation of the system or prototype can begin. The use of simple mock-ups created with readily available materials means that users can become involved in this process at a very early stage. It is important to realise that this is not simply an opportunity to present software designs to a passive audience. Rather the users should be encouraged to study and use the prototype and comment on any concerns that they may have. The various methods for user-based observation and performance measurement as well as those coming under the heading of subjective assessment can be used for this purpose.

Users should normally work with the system to accomplish tasks which should have an identifiable outcome. If particular aspects of the software are not yet present then the tasks should be designed so as to prevent users becoming reliant on the missing functionality. Evaluation sessions usually entail briefing and de-briefing components in addition to the time taken to attempt the task scenarios. The evaluation should be carried out in accordance with the evaluation plan which will specify how the data are to be collected.

The results should be treated as an integral part of the development process for the new system. The results can be used to:

- obtain design feedback, for instance, to discover usability problems with the product and, if necessary, prepare sample video tapes or event logs

- compare the results with set criteria to decide whether the objectives have been met. Statistical tests can also be used to find the probability that the results from the sample of users are representative of the whole population of users.

- reporting the results and recommendations for change

To demonstrate that the results of an evaluation are valid a description of the assumptions made about the context as well as details of the evaluation and analysis procedure should be available. Such information is typically contained in an evaluation report where the process that was carried out needs to be documented. This also confirms that a representative group of users took part and that the tasks and conditions were realistic. Naturally the testing, data collection and analysis methodology should be appropriate for the particular objectives.

Where generated a usability report may take a number of forms. This depends on the original purpose of the evaluation. If the primary aim was to capture design problems then in addition to the aspects outlined above the report should provide timely feedback in a style which helps developers improve their designs. If the software application was assessed against specific standards then the report needs to indicate which standards were applied and confirm that the assessor was qualified to apply these. It should also be clear that the software was tested in sufficient detail. Any non-conformances should be highlighted and possible ways of dealing with these suggested, unless there are good reasons for deviating from the standards, in which case the reasons should be documented. If the evaluation was designed to provide an indication of the usability of the software in performance terms then the report should also present a definition of the usability requirements (derived from contextual information) along with an indication of the status (i.e. version) of the software at the time of testing. All measurements should be explained, as should the results of any statistical analysis. Finally the report should clearly indicate whether the software as tested satisfies the particular requirements and objectives.

- iterate this activity until design (and usability) objectives are met.

Due to the ease and relative low cost of this activity it is possible to carry out several iterations of the process in order to achieve the particular design or usability objectives. Subsequent iterations may also call upon increasingly sophisticated prototypes as the mock-ups move from paper visualisations of example screens to versions of the software running on the target computing platform. The process may also shift from a relatively informal assessment of the prototype in which the views of the user are gathered to more formal evaluations in a realistic context where the user is not helped or interrupted by the evaluator. On a final note it is especially important when repeating the process that adequate records are kept to reflect changes between different versions of the design.

- tracking changes, maintenance and follow-up.

After recommendations for change have been made it is often of benefit to have continued usability input to help ensure that user requirements are incorporated into the evolving design. It is important that those who are responsible for commissioning new software or developing it themselves remain aware of the importance of usability issues.

2. Introduction to usability methods

2.1 How do you carry out user-centred design activities?

This chapter introduces a range of methods which can be used to achieve the goals of user-centred software development. Although not an exhaustive account of all the methods currently practised, those which are described in detail in this handbook represent established approaches which have been successfully applied within industry.

However even these established approaches do not have to be followed slavishly. They provide a framework - examples which can be modified with care (taking into account the common errors described for each method). In addition to these methods numerous other techniques exist and a few of those of potential interest are summarised in the last section of this chapter along with pointers towards sources of further information for the interested reader.

2.2 Overview of the methods described in chapter 3

Planning

Usability planning: effective user-centred design requires that a usability plan is constructed. For each feature in which an investment is to be made in usability work, one or more tasks are identified. For each task a task manager is appointed, an appropriate design and/or assessment method is selected and a suitable schedule is specified.

Cost-benefit analysis: a generic framework for identifying the costs and benefits associated with user-centred design activity. The first step is to identify the likely benefits to be gained by improving the usability of a system. Relevant benefits include increased user productivity, decreased user errors and decreased training costs. These benefits are assigned estimated values to indicate possible savings. The costs of the proposed usability effort (in terms of personnel and equipment overheads) must also be taken into account and subtracted from the total projected benefit value to determine the net gain to be made. The output can feed into a usability plan.

Usability context analysis: a structured method for eliciting detailed information about a product and how it will be used, and for deriving a plan for a user based evaluation of a product. Stakeholders attend a facilitated meeting to help complete a detailed questionnaire. The information collected provides details on the characteristics of the product's users, their tasks and their environment. It also lays the foundation for later evaluation activities.

Guidance and standards

ISO 9241 applicability workshop: a participatory technique in which designers attend a workshop with analysts and HCI specialists (who act as facilitators) to examine the ergonomic issues associated with their systems and scope the work required to develop solutions based on the contents of the ISO 9241 standard.

ISO 9241 software conformity assessment: this audit style method involves the assessment of a software product for conformance to the applicable requirements of ISO 9241, Ergonomic Requirements for Work with Visual Display Terminals (VDTs).

Expert-based evaluation and inspection

Heuristic evaluation: also known as expert evaluation, is a technique used to identify potential problems that operators can be expected to meet when using a computer or a telematics application. Analysts evaluate the system with reference to established guidelines or principles, noting down their observations and often ranking them in order of severity. The analysts are usually experts in human factors or HCI, but others, less experienced have also been shown to report valid problems.

Usability walkthrough: users, developers and usability specialists review a set of designs individually, and then meet to discuss each element of the design in a walkthrough meeting. Problems and their severity are identified and noted.

CELLO - evaluation by inspection: this method has parallels with heuristic or expert evaluation but the experts collaborate to a greater extent to evaluate the system in question.

Early prototyping

Paper prototyping: Designers create a paper-based simulation of interface elements (menus, dialogues, icons etc.) using paper, card, acetate, and pens. When the paper prototype has been prepared a member of the design team sits before a user and "plays the computer" by moving interface elements around in response to the user's actions. The difficulties encountered by the user and their comments are recorded by an observer and/or audio/video tape recorder.

Video prototyping: this is a variant of paper-prototyping that makes use of video equipment to create short movies of the paper interface as the elements are moved and changed by members of the design team. End-users do not interact directly with the paper prototype but can later view the video representation. This approach can be useful for demonstrating interface layout and the dynamics of navigation - particularly to larger audiences.

Computer-based prototyping: this approach to prototyping utilises computer simulations to provide a more realistic mock-up of the system under development. As such the representations often have greater fidelity to the finished system than is possible with simple paper mock-ups. Again end-users interact with the prototype to accomplish set tasks and any problems which arise are noted.

Wizard-of-Oz prototyping: this method is a variant of computer-based prototyping which involves a user interacting with a computer system that is actually operated by a hidden developer - referred to as the “wizard”. The wizard processes input from a user and simulates system output. The approach is particularly suited to exploring design possibilities in systems which are demanding to implement such as intelligent interfaces which feature agents, advisors and/or natural language processing.

Usability testing and performance measurement

User-based observation - to obtain design feedback: this is a relatively quick and cheap way to conduct a user-based evaluation of a working system or prototype with the emphasis on the acquisition of design feedback information. A small number of participants work with system while an observer makes notes. The technique can be used to identify the most significant user-interface problems, but it is not intended to provide reliable metrics.

User-based observation - to obtain metrics: this method is specifically aimed at deriving metric data and as such represents a more rigorous form of the preceding method. The real world working environment and the product under development are simulated as closely as possible. Users undertake realistic tasks while observers make notes, timings are taken and video and/or audio recordings made. The observations are subsequently analysed to derive metrics. Problems are also identified.

Co-operative evaluation: users employ a prototype as they work through task scenarios. They explain what they are doing by talking or ‘thinking-aloud’ and this is recorded on tape and/or captured by an observer. The observer also prompts users when they go quiet, and actively questions the users with respect to their intentions and expectations.

Supportive evaluation: a participatory form of evaluation akin to a ‘principled focus group’. Users and developers meet together and the user representatives try to use the system to accomplish set tasks. The designers who observe can later explore the issues identified through a facilitated discussion. Several trials can be run to focus on different system features or different versions of the system.

Subjective assessment

SUMI (the Software Usability Measurement Inventory): is a prime example of the use of questionnaires to collect subjective feedback. Following the use of a system people fill in a standardised 50-statement psychometric questionnaire and their answers are analysed with the aid of a computer program. As well as a global assessment of usability SUMI data provide information on: perceived efficiency, affect (likeability), control, learnability, helpfulness; and an assessment of how these results compare with results for similar software (deduced from a database of past results).

Cognitive workload: measuring cognitive workload involves assessing how much mental effort a user expends whilst using a prototype or deployed system. For example this can be obtained from questionnaires such as the Subjective Mental Effort Questionnaire (SMEQ) and the Task Load Index (TLX). The SMEQ has one scale which measures the amount of effort people feel they have invested in a given task. While the TLX has six scales (mental, physical, temporal, performance, effort and frustration) to measure the individual's perception of what a task has asked of them. It is also possible to collect objective data from heart and respiration rate.

Focus groups: bring together various stakeholders in the context of a facilitated but informal discussion group. Views are elicited by the facilitator on topics of relevance to the software product being evaluated. Focus groups can be used to identify initial requirements and also serve as a means of collecting feedback once a system has been in use. Several focus groups can be run to concentrate on different topics or to include the views of different sections of the user community.

Individual interviews: are a quick and cheap way to obtain subjective feedback from users based on their practical experience of a product. The interviewer may base his/her questions on a pre-specified list of items or allow the user to freely provide their views, these are referred to as structured and unstructured interviews respectively. Alternatively a combination of these approaches can be practised which results in a semi-structured interview. Interviews are often employed in conjunction with some of the other methods discussed in this handbook - particularly where de-briefing sessions take place.

2.3 Structure of the descriptions

Each method in chapter 3 is described with the following set of headings:

- **What is the method and when can it be used?**
  This provides a summary of the nature of the method, and an indication of when it can be used.

- **Contribution to design**
  This provides a concise summary of the contribution to the development process made by the particular method. Where the method overlaps with others of a similar nature it also highlights important differences.

- **Deliverables**
  Each deliverable or expected output from the method is itemised.

- **Benefits and limitations**
  First the benefits are listed and then the limitations. These details contribute to the cost-benefit analysis described in chapter 4.

- **What you need**
  An approximate estimate is given of the people, effort and the materials and other physical resources required to apply the method. All the methods may be applied with varying degrees of rigour and under circumstances of differing complexity. Hence all time and resource estimates are broad ranges to indicate the shortest as well as the longest time that may be required when applying the method.

- **How to apply the method**
  The key activities that are to be performed are listed under the sub-headings: preparation, conduct and conclusion. This provides a procedural overview for each method.

- **Quality issues**
  This describes the criteria for successful application of the method, the minimum procedure required and common errors. These are important when judging whether a method has been applied appropriately.

- **Further information**
  This section lists sources for obtaining a more detailed description of the method.

2.4 Other methods not detailed in this handbook

As well as the methods introduced above there are other approaches which are not detailed in this handbook which may nevertheless be worth consideration. These methods have been excluded from the descriptions contained in chapter 3 either because they lack a track record of industrial usage or because they overlap closely with one or more of the methods already chosen for inclusion. The following summaries provide an overview of some of these alternatives as well as pointers to sources of further information.

MUSE

The method for usability engineering (MUSE) developed at the Ergonomics Unit of University College London offers a comprehensive framework for usability activities. The primary emphasis of this framework is on initial requirements capture and design specification although MUSE can also be used to support later evaluation work. The method consists of three main phases. The initial phase focuses on information elicitation and analysis in order to capture requirements and analyse tasks. This is achieved by scrutinising existing systems and creating generalised task models. There then follows the design synthesis phase where the conceptual design is derived from a statement of user needs and models of the system and user's tasks. Finally the design specification phase addresses functional definition and user-interface design through more explicit models of the interaction, interface and display design.

Among the strengths of the MUSE approach is the importance attached to requirements capture and specification and the stress placed on early usability input. MUSE also makes an attempt to integrate the output from usability methods with conventional structured software engineering. Unfortunately, the method stresses the explicit design and documentation of a prototype rather than the exploration of design options as in a rapid prototyping model. This is at odds with the need for iteration stressed in the first chapter of this handbook and in the ISO 13407 standard. It also seems to require significant notational skills that can be time consuming to learn and apply. Furthermore, there is little evidence of its wider use in industry. However, it is quite possible that the method will benefit from increasing industrial application in the future. As such this is a methodological framework which appears to be a promising area for further development and validation. Specific details on this method can be found in:

Lim, K.Y. and Long, J. (1994). The MUSE method for usability engineering. Cambridge University Press.

QUIS

The Questionnaire for User Interface Satisfaction (QUIS) is a subjective assessment technique. This questionnaire focuses directly on the user interface and was developed by Shneiderman and colleagues in the late eighties. It consists of 5 sections where the first assesses the users overall reactions to the software and the other 4 assess: screen, terminology and system information, learning, and system capabilities.

Despite the popularity of this approach to subjective assessment, particularly in America, there appear to be some problems associated with the questionnaire. The study by Chin et al (1988) suggests that some of the QUIS items may not be independent and thus the different sections of the questionnaire may fail to capture different aspects of the end-user's experience. Furthermore, and in contrast to the SUMI questionnaire, independent data on the statistical reliability of QUIS does not appear to exist. QUIS is also not supported by the type of normative data which is supplied with the SUMI questionnaire.

Further information on this method can be found in:

Chin, J.P., Diehl, V. and Norman, K.L. (1988). Development of an instrument measuring user satisfaction of the human-computer interface. Proceedings of CHI'88, May 15-19, Washington, pp. 213-218.

Shneiderman, B. (1992). Designing the user interface: Strategies for effective human-computer interaction, 2nd edn, Reading, MA: Addison-Wesley.

Checklists and Guidelines

The use of checklists represents a form of inspection-based assessment which can be practised by experts or in some cases by end-users. A checklist contains a list of issues which are used to identify weaknesses or problems with a system. An early example of this form of assessment was described by Ravden and Johnson (1989) and featured a detailed checklist based on nine criteria such as consistency and visual clarity. This particular checklist could be used by actual end-users although not all the questions are suitable for modern GUI systems. Checklists of ergonomic criteria are central to expert-based methods and as such are covered in chapter 3, however further information can also be found in:

Ravden, S. J. and Johnson, G. I. (1989). Evaluating usability of human-computer interfaces: a practical method. Chichester, England, John Wiley & Sons.

Another basis for usability design and assessment stems from the numerous sources of guidelines such as standards and styleguides. These are discussed in more detail in chapter 5.

3. Individual method descriptions

The methods are organised into 6 categories:

- Planning
- Guidance and standards
- Early prototyping
- Expert-based evaluation
- User performance evaluation
- Subjective assessment.

3. 1. Planning

The following collection of methods provide an important foundation for later usability work. For example the Usability Context Analysis method should be viewed as a necessary precursor to evaluation activities, in order to ensure that all relevant contextual details have been taken into account when planning an assessment exercise and also as an aid to interpreting the obtained results. These methods, which support the preliminary user-centred design activities described in the previous chapter, are:

- Usability planning
- Cost-benefit analysis
- Usability context analysis.

3.1.1. Usability planning

What is the method and when can it be used?

This technique is a means of defining and managing the user-centred design activities that will take place during the development of a product or system. Often a cost benefit analysis of user-centred design work is carried out and its results summarised in the first part of the usability plan. For each feature in which an investment is to be made in user-centred design work, one or more tasks is identified. For each task a task manager is appointed, an appropriate technique is selected and a schedule is specified. The usability plan is a living document, and undergoes regular reviews as the project progresses. Usability planning should be carried out for all user-centred design projects.

Contribution to design

Usability planning provides an important means of managing individual usability activities as well as the overall role played by usability input within a software engineering programme. This defines the scope of usability work required to produce a design that is efficient, effective and satisfying in use.

Deliverables

A usability plan report is created. This details the scope of the intended usability activities and defines the actions and timescales required to implement these.

Benefits

- Enables priorities to be assessed and facilitates the efficient allocation of resources.
- Ensures that usability work is carried out in a co-ordinated rather than ad-hoc fashion.
- Provides clear visibility of what usability work is going on and what the overall aims are.

Limitations

- Usability plans may not be stable because of continually changing project plans and software development schedules.
- Usability plans are also constrained by the need to deliver the results in sufficient time for them to be acted upon by designers and implementors.

What you need

Prior input

As a part of the usability planning exercise a cost-benefit analysis (see the following method entry) may be required to indicate that it is worth making an investment in further user-centred design activities. Also note that the output from a context analysis meeting can provide a valuable input to any usability plan.

People and effort

Between 2 and 5 people can be involved in usability planning. One of these should have human factors experience while between 1-4 key project personnel may also be involved. The usability representative will require at least 4 days of available effort while other members of the project will be occupied by this process for up to 2 days. This includes time for preparation, holding meetings, defining and reaching agreement on a plan and formalising the plan in a report.

Materials and physical resources

A meeting room will be required to discuss the details of the plan.

How to apply the method

Preparation

Become familiar with the features for which cost benefit analysis has indicated it is worthwhile making a user-centred design investment.

Conduct

Hold a meeting with the key project personnel who have an interest in the usability of the product. For each selected feature or aspect of the system discuss what sort of usability targets could be set, the work that needs to be done to achieve those targets, and the most appropriate methods to assess the developing system. Following this create the usability plan. For each selected feature specify a programme of work which will deliver the desired target. If a detailed target has yet to be defined, create a task to do so. A person who is responsible for carrying out the usability work should be identified along with the resources that will be required, and a schedule with milestones. Get agreement to each part of the plan with the relevant people.

Conclusion

Document the plan in a usability plan report. Implement the plan as specified and update where necessary.

Quality issues

Criteria for successful application of the method

All relevant parties should be consulted when constructing a usability plan and their backing should be secured. The plan will need to be sufficiently flexible to accommodate changing development schedules.

Minimum procedure required

All the key steps outlined above should be carried out.

Common errors

Failing to take account of all possible contingencies or choosing inappropriate methods given the aims of the plan.

Further information

Planning user-centred activities is discussed in the forthcoming ISO 13407 standard, as well as being summarised in the first chapter of this handbook.

3.1.2. Cost-benefit analysis

What is the method and when can it be used?

This method description presents a generic framework for identifying the costs and benefits associated with any user-centred design activity. The first important step is to identify the likely benefits to be gained by improving the usability of a system. Relevant benefits include increased user productivity, decreased user errors and decreased training costs. These benefits are then assigned estimated values to indicate possible savings. The costs of the proposed usability effort (in terms of personnel and equipment overheads) must also be taken into account and subtracted from the total projected benefit value to determine the net gain to be made.

For organisations already committed to user-centred design a cost-benefit analysis is not essential but it can provide a valuable input when formulating a usability plan. The technique can be used repeatedly as a development project progresses to reassess the importance of various activities. The process can also be used to compare different usability methods and so aid selection, a topic which is discussed further in chapter 4.

Contribution to design

A cost-benefit analysis can be used to determine which aspects of the system will benefit most from improved usability, and which usability methods will prove most cost-effective in delivering those benefits.

Deliverables

A report identifying the benefits that could accrue from specified improvements to the system, and estimating the user-centred design investment required to achieve these benefits is usually generated.

Benefits

- Cost-benefit analysis enables user-centred design effort to be directed towards those aspects of system usage which promise the best return.
- The method provides data to make a case for investment in user-centred design work, particularly in terms of the cost savings to be made during the product's life.
- The method helps identify the goals to be achieved during the design / evaluate / redesign process.

Limitations

- Cost-benefit analyses depend upon estimation and prediction, so there can be a large margin of error.

What you need

People and effort

Up to 3 people can be involved in a cost-benefit analysis process; 1 analyst, 1 project manager and ideally 1 user manager. The method can be applied with as little as 5-6 days effort (for the three participants) or may require up to 20 days for a complete analysis of a complex product or process. These figures cover the preparation time (for the analyst to become familiar with the project), meeting time (for the participants to identify the benefits of interest, the implications for savings and the level of usability investment required), and time for analysis and reporting.

Materials and physical resources

A room to hold the meeting in will be required, and access to the system under development may also be needed.

How to apply the method

Preparation

The parties involved in the cost-benefit analysis process meet at an agreed place and time and then proceed by creating a shortlist of potential benefits to be gained by improving the usability of the system in question. As previously indicated this might include: increased user productivity, decreased user errors, decreased training costs, as well as decreased user support costs and savings gained by making changes early on in the design life cycle.

Conduct

Having identified these potential benefits, assumptions must then be made in order to estimate the potential value of each benefit. For illustration purposes an example based on decreased training costs will be used. Consider a scenario where 250 staff are to receive training in the use of the new system. For the sake of argument assume that training without the benefits of user-centred design would require 3 days and that this can be reduced by 3 hours because of effective usability engineering. Also assume a fully loaded hourly income of £25. Based on these figures the following calculation can be carried out, which provides an estimate of the savings to be made compared to not carrying out any usability work:

250 (staff) x 3 (hours) x £25 = £18,750

Of course the cost of the usability input (based on the overheads of the particular methods used) must be subtracted from this to identify the net gain as well as the ratio of costs to benefits. Further calculations can be carried out for the other types of benefits where applicable and the potential savings totalled to offset the costs of usability input. It should also be noted that with the exception of the training benefit most of the other benefits can be accrued for each year that the system is in use.

Conclusion

Summarise the results of the analysis in a report, and disseminate to all interested parties. These results may have implications for a usability plan and the selection of user-centred design activities and these should be followed up where necessary.

Quality issues

Criteria for successful application of the method

To carry out this form of analysis reliably a sound knowledge of both the system and the processes used to develop the system is required as is a thorough appreciation of the task context. Awareness of usability methods and particularly there costs is also important.

Minimum procedure required

The key elements of the generic process are outlined above. Further analyses are possible to provide such terms as payback period and net present value. These are not essential - but are discussed in detail in Bias and Mayhew (1994) for the interested reader.

Common errors

Under or over-estimating the importance of particular benefits. Selecting an inappropriate method for achieving these benefits. Failing to update the analysis as the project circumstances change.

Further information

The general topic of cost-benefit analyses as well as more complex strategies for accomplishing this are discussed in the following title:

Bias, G. and Mayhew, D. (Ed’s.): Cost-Justifying Usability. Academic Press, 1994.

3.1.3. Usability context analysis (UCA)

What is the method and when can it be used?

Usability Context Analysis (UCA) is a structured method for eliciting detailed information about a product and how it will be used, and for deriving a plan for a user based evaluation of a product. In this method stakeholders attend a facilitated meeting to detail the actual circumstances (or intended use) of a product. This is summarised in a document called the Context Report Form, which is then examined by a usability consultant who decides if each factor is indeed important for the usability of the product. Following this inspection a summary list of these factors, called the Context of Evaluation, is produced. This list specifies important characteristics of the products' users (or groups of users), their tasks, their environment, and also lays the foundation for an observational evaluation.

Context Analysis meetings should take place as early as possible in the design of a product. However the results of these meetings can be used throughout the lifecycle of the product; being continually updated and used for reference. The method is also of general relevance to a wide range of systems and processes. The objective of the method is to collect information - this should be done preferably through a well-prepared meeting, but it may also be done through a series of personal interviews, if for instance schedules in a busy organisation do not permit essential personnel to attend a general meeting.

Contribution to design

This method can contribute to the design process by ensuring that both design and evaluation activities are conducted with due regard for the characteristics of the intended users, their tasks and their environment. Context meetings are also a particularly effective way of making user characteristics explicit for developers.

Deliverables

A Product Report - a two page description of the product in a standard format.

A completed Context of Use questionnaire which summarises the characteristics of the users, their tasks and their environment.

A Context of Evaluation document - a summary list of those factors in the questionnaire considered important for the usability of the product.

An Evaluation Plan (optional) for a user-based evaluation.

Benefits

- The method offers a framework to ensure that all factors which may affect usability are identified before design work starts.
- Context meetings bring together all the people relevant to a development programme, early on in the project.
- It also helps to ensure that evaluation activities produce valid results, by specifying how important factors are to be handled in an evaluation, and by defining how well the evaluation reflects real world use.
- For comparative evaluations the method documents the circumstances of each evaluation (e.g. for comparisons between novice and expert groups of users).

Limitations

- The success of this method depends upon competent chairing to keep the meeting focused on the relevant issues. Familiarity with the Context of Use questionnaire by the chairperson is essential.
- Context meetings can be difficult to arrange because of the number and type of people usually involved.
- Context meetings can be frustrating without competent chairing, and the key issues can be hard to identify.

What you need

People and effort

Up to 10 people can be involved: 2 analysts (one acting as a facilitator and one recording the information provided during the meeting) and up to 8 stakeholders. The latter group typically consists of representatives from the development team, project managers and the end-users themselves. A context meeting will occupy the analysts for approximately 3 days (1 day for the meeting and 2 days for the preparation beforehand and then writing the context report). The stakeholders will be required for 1 day or less depending upon the length and complexity of the context meeting.

Note: an evaluation plan will take an additional 1 - 3 days to create should this be required.

Materials and physical resources

No specific hardware or software is required for this method although a PC projector can enable the questionnaire to be filled in and displayed during the context meeting. A word processor is needed to generate a report. The facilitator of a context meeting should also be familiar with the system to be evaluated and so may require access to a functional prototype or a finished product.

All participants should be provided with a written description of the Context Analysis process and a blank copy of the context of use questionnaire, before the context meeting. The people supporting the production of the evaluation plan should be given a written description of the evaluation planning process. Finally a room large enough to accommodate all the participants will be required.

How to apply the method

Preparation

When applying this method it is important that the relevant stakeholders are present, so the first step is to identify all relevant individuals who will be involved in the process and the context meeting. Having identified the people you want to take part, a room where the context meeting can be held should be located, and a suitable date arranged. Write to all participants with details of the date, place and agenda, and include a copy of the Product Report and a description of the context analysis process. If possible complete the Product Report Form in conjunction with the user representative(s). Also go through the Context Report to identify in advance any factors which do not need to be discussed at length during the meeting itself.

Conduct

At the meeting, give each participant a blank questionnaire and work through this filling in the 'Context of Use' column. This will provide details on the intended users of the product, their tasks and the conditions of their working environment. Careful chairing will need to be practised to ensure that this information is obtained.

Conclusion

After the meeting, identify the critical components which may affect usability and document them in a Context of Evaluation document. If an evaluation is to be performed, produce an evaluation plan detailing the users, the tasks they will carry out, and the circumstances of the evaluation. This plan will also need to describe how the critical components of the context will be handled.

Quality issues

Criteria for successful application of the method

Effective facilitation of the context meeting by the chairperson is crucial, training or prior experience in this process is highly recommended, and familiarity with the Context Report Form is essential. One should also ensure that all the relevant stakeholders are present.

Minimum procedure required

The steps described above are all necessary for the satisfactory application of the method.

Common errors

Allowing the context meeting to drift from its main objectives. Failing to account for the context of use when planning and running an evaluation.

Further information

A description of the Context Analysis process is contained in ‘Usability Context Analysis: a Practical Guide’, from NPL Usability Services, (National Physical Laboratory), Teddington, Middlesex, TW11 OLW.

Training in conducting Context Analysis is also provided as part of the National Physical Laboratoryís Performance Measurement Training Course (over 2 days).

3.2. Guidance and standards

The next methods to be described are designed to ensure that the knowledge and advice provided by relevant human factors guidance and standards documents is incorporated into the development of interactive systems. These sources provide a wealth of information to inform design activities, however much of the advice can be difficult to apply in a practice. The following methods counteract this problem. They are:

- ISO 9241 applicability workshop
- ISO 9241 software conformity assessment

3.2.1. ISO 9241 applicability workshop

What is the method and when can it be used?

This is a participatory technique in which designers attend a workshop with analysts and HCI specialists (who act as facilitators) to examine the ergonomic issues associated with the system being developed and scope the work required to develop solutions based on the contents of the ISO 9241 standard. This standard contains the best and most widely agreed body of software ergonomics advice. In particular the processes recommended in Part 1 and Annex 1 of parts 12-17 of the standard ensure a systematic evaluation of each clause to check its applicability to the particular system(s) under consideration. The combination of these processes and recommendations is used to ensure that the principles of software ergonomics have been considered in the development of a system. This approach supports (and may supersede) the use of style guides.

Contribution to design

This technique can be particularly effective at helping designers, who may be unfamiliar with human factors and usability issues, determine what is required to make their system compliant with the ergonomic principles contained within the ISO 9241 standard. The guidance document produced by the workshop helps designers to incorporate relevant usability requirements into their specifications and designs.

Deliverables

A guidance document detailing relevant ergonomics issues is provided for the project. This is part of the specification for the system and the basis for the 'look and feel'. It contains an analysis of the ergonomic recommendations which the design team need to take account of during the design and also lists: the extra work required in order to collect supporting information and the methods to be used to collect this information.

Benefits

- End users will be using a system based on a design embodying best practice in software ergonomics.
- This approach encourages buy-in to ergonomics by involving the designers in creating the specification document which they will work to and at the same time teaching them how to interpret it.

Limitations

- Advice is limited to the scope of the standard. Further work will be required for designers to develop the detailed system in line with the recommendations.
- It should be noted that most parts of the standard are in draft form, but all are essentially technically correct and usable.

What you need

People and effort

Up to one week of usability analyst / facilitator effort is required. This person should be familiar with using style guides and standards and human factors knowledge is helpful. The level of expertise required for follow-up activities is defined in the Standard although the follow-up effort is variable. If the project has a design cycle which usually collects information for design then the extra effort is small. The number of other people involved varies depending upon the attendance level for the workshop, but typically ranges from 2 to 8 individuals.

Materials and physical resources

The main documents are the parts of the Standard, and these may have to be purchased in the appropriate language.

How to apply the method

Preparation

Participants need to have either a specific system under development or be in the process of defining a project or house style. The training itself does not require end users to be present although this option could be considered. A context of use statement is required. The client should send details of the development process to the facilitator. Specific or typical requirements should be known and checked. Just prior to design is the best time to start the workshop.

Conduct

The workshop commences with an introduction to the standard, the context of use and the system/organisation requirements. There then follows a review of the requirements for the system to help identify the relevant parts of ISO 9241. The attendees split into groups and each group tests the applicability of the recommendations: in allocated parts and against allocated viewpoints then produces a list of applicable and possibly applicable recommendations. The groups then reconvene and agree design recommendations for the specified level of detail of the system. Appropriate evaluation methods for checking applicability are identified if necessary. A style guide of clauses or specifications is then developed at the appropriate level of detail. The process is terminated when the design or guide is complete. This guide is also revised if the context of use, the usersí tasks or the Standard changes.

Conclusion

Further work with end users, task analysis or trials of exploratory prototypes, may be recommended after the applied workshop. A need for detailed task analysis and more detailed user requirements may also be identified as a result of the workshop. If a project has an ISO 9000 quality system, standards and their interpretation will be handled by this system.

Quality issues

Criteria for successful application of the method

This approach depends very much upon the thoroughness applied when considering the conditional parts of the recommendations. Between two and six attendees is manageable, greater than six will degrade the quality of the meeting and the maximum to be considered should be eight. Planning and facilitation should be sympathetic to the clientís needs. The facilitator should understand the clientís development method and system.

Minimum procedure required

Most of the steps described above should be carried out although selecting usability evaluation methods may not be required.

Common errors

Lack of belief in a user-centred approach in the design team. Hidden assumptions and team bias about the stakeholders and the capabilities of the technology. Neglecting to take full account of the usersí context and tasks can also be a problem. Lack of communication amongst attendees. Not taking into account all viewpoints and elements of the system. Trying to cover the system in too many levels of detail at one time.

Further information

Primary reference sources: ISO 9241 Parts 1,2, 10-17. Guidelines on Safe Work with VDUs. LR Capability Statement and reports of ISO 9241 common interest group meetings.

Contact Jonathan Earthy, tcsjve@aie.lreg.co.uk, Lloyd's Register House, 29 Wellesley Road, Croydon, CR0 2AJ, UK, tel +44 181 681 4040, FAX +44 681 4839 for further details.

3.2.2. ISO 9241 Software conformity assessment (SCA)

What is the method and when can it be used?

This method provides the basis for assessing a software product for conformance to the relevant requirements as detailed in the ISO 9241 standard: Ergonomic Requirements for work with Visual Display Terminals (VDTs). Developers provide documentary evidence regarding their processes and one or more auditors examine these documents and visit the site to interview relevant staff. The assessors then determine if a certificate of conformance is warranted or if not feedback is provided on non-conformances.

Contribution to design

The method ensures that a product conforms with ISO 9241 and thus embodies good ergonomic principles.

Deliverables

A statement of the scope of the assessment, a certificate of conformance, or, a list of non-compliances relating to the product and the development process is produced.

Benefits

- Assurance that a software product meets the explicit and implicit requirements of the ISO 9241 standard.
- Also gives an assurance of the system meeting clause 19 of Directive EU 90/270
- Conformance testing identifies errors in the software ergonomics of the product and consequently - usability defects.
- Application of the method helps to ensure the quality of in-house work and shows tangible benefits for a wider process improvement programme.

Limitations

- Much of the standard is currently only available in draft form. However, most parts will be published over the course of the coming year.
- Although conformance to ergonomic principles suggests a high likelihood of product usability it does not constitute a guarantee.

What you need

People and effort

At least one human factors expert is required to act as the auditor, and others may be needed depending upon the size and complexity of the project. The application of this method can be expected to take up to 2 weeks with scoping, planning and managing the activity occupying up to 5 days of effort and another 5 days being spent on-site and reporting the results. The actual amount of time spent on site again depends on the size of project. The organisation or group whose project is being assessed should also supply a member of staff to assist the auditor(s).

Materials and physical resources

A variety of documents can be consulted when applying this method. This might include project quality documents, development documentation and standards, context of use information, user requirements, ergonomic requirements, completed applicability and adherence checklists, configuration management documentation, evaluation documentation and other supporting documentation. In addition a computer running the system to be assessed will be required, plus a CASE or project database if necessary.

How to apply the method

Preparation

The system developer defines and follows a development process which takes account of the context of use for the product or system, the userís tasks and any relevant ergonomic requirements. The developer assesses which parts of the standard and which recommendations apply, applies them and then assesses whether the resulting product adheres to the recommendations. The developer then retains evidence of having followed this process, and an indication of the applicability and adherence. The lead auditor then defines the scope of a study and assessment checklist. Auditors subsequently study the project documentation that is provided by the developer.

Conduct

Auditors visit the developerís site and examine the evidence of adherence to the procedures described in the documentation. This is carried out through a programme of inspections and interviews with the developerís staff. Repetition of some evaluations may be required by the auditors. Auditors then present their findings to the developer.

Conclusion

If the procedures and evaluation results comply with those defined by the developer and if the procedures comply with the requirements of the 9241 Standard and the SCA a certificate is awarded within the terms of SCA. If the procedures or evaluation results do not comply with the relevant criteria in the 9241 Standard or the SCA then notices describing the non-compliances, and remedial actions are issued to the developer. After performing remedial actions the developer may apply for re-examination.

Quality issues

Criteria for successful application of the method

Effective planning and management of the assessment is crucial. The assessor should understand the clientís development method and system. Assessor impartiality, professionalism, confidentiality, and freedom from external pressures and incentives is also important.

Minimum procedure required

The scope is decided at an initial meeting. The less the developer has done to take account of the standard in development the more the assessors have to do to prepare for and carry out the assessment and the greater the cost of conformance assessment.

Common errors

Hidden assumptions and assessor bias may creep in. Ignoring contextual issues and tasks should also be guarded against. Poor communication between assessors and informants can also lead to difficulties.

Further information

Cost of assessment depends on the degree to which the client has followed the processes and recommendations described in the Standard.

Primary reference sources: ISO 9241, LR Software Conformity Assessment System: Procedure SC94, LR SCA 9241 module, reports from LR ISO 9241 exploitation workshops, TickIT training course notes.

Contact Jonathan Earthy, tcsjve@aie.lreg.co.uk, Lloyd's Register House, 29 Wellesley Road, Croydon, CR0 2AJ, UK, tel +44 181 681 4040, FAX +44 681 4839 for further information.

3.3. Early prototyping

Early prototyping methodologies permit the exploration and evaluation of different design options from the initial stages of development through to the latter stages as well. There are two general approaches which make use of either simple paper materials or computer systems to create the prototype. These result in prototypes of differing sophistication although variants of both approaches will provide useful information. These activities provide the foundation for user-based observations and support the collection of design feedback and metric data. The early prototyping techniques which are discussed here include:

- Paper prototyping
- Video prototyping
- Computer-based prototyping
- Wizard-of-Oz prototyping

3.3.1. Paper prototyping

What is the method and when can it be used?

This method enables draft screen designs to be very rapidly simulated and tested. Members of the evaluation team create a paper-based simulation of interface elements (menus, dialogues, icons etc.) using paper, card, acetate, and pens. When the paper prototype has been prepared a member of the team sits before a user and “plays the computer” by moving interface elements around in response to the user’s actions. In this way the motion of a mouse pointer can be simulated and menus and commands can be activated by moving and replacing key elements. A facilitator provides task instructions and encourages the user to express their thoughts and impressions. If other observers are present they may make notes of what happens. The technique is particularly suited to the early stages of the design cycle. To increase user involvement some variants of the method have the users and designers collaborate to build the paper prototype.

Contribution to design

Design options can be explored at an early stage with little cost - thus supporting an iterative development process.

Deliverables

A report detailing the design that was built and the rationale behind it; the problems identified during the evaluation; and recommendations for the next phase of design is produced.

Benefits

- Potential usability problems can be detected at a very early stage in the design process before any code has been written.
- Communication between designers and users is promoted.
- Paper prototypes are quick to build / refine, thus enabling rapid design iterations.
- Only minimal resources and materials are required.

Limitations

- Because of their simplicity, paper prototypes do not support the evaluation of fine design detail.
- This form of prototype cannot reliably simulate system response times or be used to deliver metric data.
- The individual playing the role of the computer must be fully aware of the functionality of the intended system in order to simulate the computer.

What you need

People and effort

Two analysts or designers will be required to apply this method - one to play the role of the computer and the other to act as the facilitator. This division of labour allows the person playing the computer to remain silent and thus maintain the illusion of the computer role. It is most cost effective to iterate several times with small numbers of users rather than run one large evaluation. 3-5 users have been shown to be most cost effective (Nielsen, 1993). User sessions should be accomplished within 2 days, the analysts / designers will require approximately 5-6 days to create the prototype, recruit users, run the sessions, analyse the results and generate a report.

Materials and physical resources

Materials which are useful to create the prototype include: paper, card, Post-It notes, adhesives, acetate sheets, scissors and coloured pens. In terms of documents, task instructions for the users will be required and a list of any questions to be asked of the user should be at hand. The prototype should also be built with reference to any relevant requirements specifications. A room will be required to conduct the evaluation and tape or video recording facilities can be used if desired.

How to apply the method

Preparation

Identify the people you want to take part, the room where you want to hold the evaluation and the most suitable date. Assemble the necessary materials. Derive the design and construct the paper prototype. Create a different piece of card for anything that moves, changes appearance or comes and goes. Document realistic scenarios and tasks for the evaluation. Plan the evaluation process and document any questions that are to be asked. Pilot the evaluation procedure and practice playing the role of the computer.

Conduct

Conduct each session by having the facilitator provide the task instructions and asking questions to explore the user's impressions and intentions. The person acting as ëthe computerí presents appropriate paper representations as the user works through the task and interacts with the "system". Observers make notes. Conduct post-session interviews with the users, drawing upon pre-set questions and any issues raised during the evaluation of the prototype.

Conclusion

Analyse the information obtained, and identify the problems experienced. Derive recommendations for improving the design. Include these details in a written report.

Quality issues

Criteria for successful application of the method

Effective use of available materials in order to simulate a system or interface. A 'computer role-player' who is familiar with the scope and functionality of the proposed system in order to provide as realistic a simulation as possible.

Minimum procedure required

All the steps described above are required.

Common errors

The role-player directs the user or simulates unrealistic system behaviour. Use of inappropriate tasks due to lack of awareness of context

Further information

Rettig, M. (1994): Prototyping for tiny fingers. Communications of the ACM, Vol.37, No.4, April 1994, pp. 21 - 27.

3.3.2. Video prototyping

What is the method and when can it be used?

This method allows designers to create a video-based simulation of interface functionality using simple materials and equipment. As with paper-prototyping interface elements are created using paper, pens, acetates etc. Video equipment is then used to film the functionality of the interface. For example a start state for the interface is recorded using a standard camcorder. The movements of a mouse pointer over menus may then be simulated by stopping and starting the camcorder as interfaces elements are moved, taken away and added. Users do not directly interact with the prototype in this approach however they can view and comment on the completed video-based simulation. This variant on paper-prototyping is particularly suited for simulating the dynamic character of a simple interface mock-up and can be used during the early stages of the design cycle to demonstrate design options and concepts to an audience.

Contribution to design

Paper mock-ups of design options can be demonstrated dynamically in order to elicit design feedback. This will help to ensure that later prototypes avoid many usability problems. However, the method cannot be used to capture a user’s interaction with the prototype system.

Deliverables

The primary deliverable is a video recording of the simulated interface which can be shown to groups of people, including end-users, in order to elicit their feedback.

Benefits

- Usability problems can be detected at a very early stage in the design process (before a commitment to code has been made).
- Provides a dynamic simulation of interface elements that can be viewed and commented on by both design teams and intended users.
- Minimal resources and materials are required to convey product feel.
- The technique can be utilised by those with little or no human factors expertise.

Limitations

- Staff familiar with the functionality of the intended system are required to create the video prototype.
- The method does not actually capture a user interacting with the prototype and lacks the interactive element of other prototyping methods. As such it would perhaps be most suited for demonstration purposes where larger audiences are involved and proof-of-concept is the goal.
- Because of the use of simple materials, video prototypes do not support the evaluation of fine design detail or the collection of metrics.

What you need

People and effort

The method calls for two people, one to manipulate the interface elements and so operate the "computer" and the other to control the camera. Little video production expertise is required, although it could be time-consuming to create more complex sequences using stop-motion animation. Between 3 and 5 days needs to be set aside to create the paper as well as the video prototype, and more time may be required depending upon the complexity of the desired results. End users are not directly involved although the video prototype is created with a view to later displaying it to audiences.

Materials and physical resources

As with paper prototyping only simple materials are required to create the elements of the prototype to be committed to video-tape. These include paper, acetate, pens adhesives etc. A video camera is also required to capture and replay a simulation and a tripod would be of benefit. Basic costs will also be incurred for video-tapes.

How to apply the method

The method parallels that described for paper prototyping in several respects, although the emphasis on creating a video-based simulation is a distinct feature, as is the absence of users who directly interact with the prototype.

Preparation

First, allow enough time to create the paper prototype, design some scenarios of use for demonstration purposes, and produce the video-based simulation. It should be remembered that even brief sequences of stop-start animation can be time consuming. Assemble the necessary materials. Construct the paper prototype, using separate stock for menus, dialogue boxes and any element that moves or changes appearance. A paper-based mouse pointer for instance can be attached to the end of a strip of acetate so that it can be moved without the operators hands appearing on the video recording. The person manipulating the interface elements should practice playing the role of the computer. Also ensure that the recording facilities are available and functioning. Ideally the camera should point directly at the prototype, perhaps by being mounted above a table where the materials are placed.

Conduct

One person should manipulate the elements of the paper prototype while another person controls the video camera. For example a menu selection can be captured by initially filming a shot of the paper desktop and subsequently filming a brief sequence as the mouse pointer is moved on a transparent arm to a menu item. The video camera is then paused while a paper representation of a menu is placed under the camera, filming then continues while the mouse pointer passes down the menu items.

Conclusion

Once completed the video-based prototype can be shown to design teams as well as intended users to obtain evaluative feedback. Several video prototypes can be created and shown to an audience in order to explore different design options. Where necessary the prototype can be refined and the above process repeated. The use of video prototypes supports the exploration of design options by providing a dynamic simulation which can be shown to both potential users and colleagues. This can result in recommendations for the refinement of the initial prototype. All findings should be fully documented in a written report.

Quality issues

Criteria for successful application of the method

Familiarity with any relevant contextual issues and requirements when designing the prototype and simulating likely usage scenarios is required. Maintaining a willingness to change the prototype in light of feedback is also important.

Minimum procedure required

If a paper-prototype already exists then 1-2 days effort can be saved by not having to create one from scratch.

Common errors

Failing to take into consideration user and task requirements. Expending too much effort to achieve a 'polished' video prototype rather than securing feedback and then iterating the prototype further.

Further information

Vertelney, L. (1989): Using video to prototype user interfaces. SIGCHI Bulletin, Vol. 21 (2), pp. 57-61.

3.3.3. Computer-based prototyping

What is the method and when can it be used?

This method supports the development and exploration of different design concepts through software prototypes. This form of prototyping has grown increasingly popular with the advent of rapid prototyping tools and development environments which make it relatively simple to create a simulation of a proposed system. Computer-based prototyping aims to reduce the iterative development cycle. Interactive prototypes can be developed throughout the development schedule and can be quickly replaced or changed in line with design feedback. This feedback may be derived from colleagues or from the experiences of users as they work with the prototype to accomplish set tasks. Within software engineering circles the method is closely associated with user interface management systems and various design support tools. The latter tools offer the designer libraries of procedures and graphical interface elements for defining the software's logical structure and 'look-and-feel'.

These kinds of prototypes exhibit a higher fidelity with the end product than those created as part of other methods such as paper prototyping. The method requires more sophisticated technical resources than is the case with low-fidelity prototyping methods which rely on paper materials. An additional cost of use is the level of human expertise required to master the supporting development tools, along with the time necessary to implement a software prototype. However computer-based prototypes can be used to obtain metric data and details of user performance using the system.

Contribution to design

This allows interactive simulations to be assessed with end-users to identify usability problems. The high-fidelity prototypes can also support the collection of performance data.

Deliverables

A report detailing any problems users have experienced with the system as well as relevant metrics is created.

Benefits

- This approach permits the swift development of interactive software prototypes.
- Prototypes created by this method have a high fidelity with the final product.
- The prototypes created under this method also support metric-based evaluations.

Limitations

- The method requires software development skills.
- Although rapid, the method is more time consuming than paper-based approaches.
- The resources required are greater due to the use of software and hardware rather than paper and pens.
- Due to the greater investment in skills and time there may be some reluctance to 'throw away' a computer-based prototype in contrast to simple paper mock-ups.

What you need

People and effort

A designer or developer will be required to construct the prototype, the time taken for this will depend upon the individual application domain and the skills of the developers. A human factors specialist will be required to run the evaluation sessions with the prototype, the time taken for this will depend on the kinds of information that is required. Further details are provided in the separate entries for user-based evaluations.

Materials and physical resources

Development tools and systems will be required to create the computer-based prototype. Access to relevant requirements and specifications documents should also be possible.

How to apply the method

Preparation

Firstly, allow enough time to create the prototype. If the prototype is to be evaluated with users then time will be required to design relevant tasks, recruit the users, evaluate the prototype and report the results. Assemble the necessary equipment, including the hardware and software tools necessary to create the interactive prototype. Develop the prototype itself. Select appropriate users to test the prototype, trying to cover the range of users within the target population. A facilitator will also be required to instruct the users and run the evaluation. Prepare realistic tasks to occupy the users as they work with the prototype. Pilot the evaluation procedure and ensure that the prototype can be used to accomplish the set tasks. Ensure recording facilities are available and functioning.

Conduct

Conduct each session. The facilitator instructs the user to work through the allocated tasks, and interact with the system as appropriate. If necessary additional information can be obtained by interviewing users following their use of the prototype. After the sessions each user should be debriefed and thanked for their input.

Conclusion

Analyse the obtained information and then summarise the observations and user evaluations. Determine the themes and severity of the problems identified. Summarise design implications and recommendations for improvements and feed these back to the design team. Video recordings can support this. Where necessary refine the prototype and repeat the above process.

Quality issues

Criteria for successful application of the method

The prototype should be developed with due regard for contextual issues and the demands of the task(s).

Minimum procedure required

All the steps described above are required.

Common errors

Inappropriate design of the prototype and the tasks due to ignorance of the context can lead to difficulties.

Further information

Widely discussed in the literature, for example see:

Hix, D. and Hartson, H.R. (1993): Developing interfaces: ensuring usability through product and process. John Wiley & Sons, pp. 249-281.

3.3.4. Wizard-of-Oz prototyping

What is the method and when can it be used?

This variant of computer-based prototyping involves a user interacting with a computer system which is actually operated by a hidden developer - referred to as the "wizard". The wizard processes input from a user and simulates system output. During this process the user is led to believe that they are interacting directly with the system. This form of prototyping is beneficial early on in the design cycle and provides a means of studying a user's expectations and requirements. The approach is particularly suited to exploring design possibilities in systems which are demanding to implement such as those that feature intelligent interfaces incorporating agents, advisors and/or natural language processing.

Contribution to design

Details on usability problems can be identified to aid in the design of novel multimodal interfaces.

Deliverables

The evaluation of Wizard-of-Oz prototypes yields information regarding user expectations and requirements for novel technologies. The data gained from this approach may be used to refine the initial prototype, and inform subsequent evaluations. These details are contained within a written report.

Benefits

- This approach allows usability requirements and issues to be explored at an early stage in the design process, particularly for systems which go beyond readily available technology. While the technique may lack the general applicability of other prototyping approaches it is particularly suited to multimedia and telematics applications.
- The member of the design team who plays the wizard can gain valuable insights from the close involvement in the user's activity.

Limitations

- The person playing the role of the wizard must appreciate the functionality of the proposed system in order to provide a convincing representation.
- This approach requires a higher commitment of resources than other approaches to prototyping such as those that rely on simple paper-based materials.
- It may be difficult to carry this out effectively in situations where there is a large graphic element in the interface.

What you need

People and effort

Two staff are required to conduct the evaluation - one to play the wizard (and thus maintain the illusion of the test system) and another to instruct the user and record the session. The wizard should have an in-depth understanding of the proposed system so that system responses are logical and not beyond the realms of possibility. The approach is likely to require practice to reach competency. The time overhead largely depends upon the task domain and the number of users exposed to the prototype.

Materials and physical resources

Two computer systems would be required, one each for the user and the wizard. Video and/or audio recording facilities will be required as well as two rooms to house the computers and the user and wizard in. All documentation relating to task instructions etc. should be made available.

How to apply the method

Preparation

Firstly, allow enough time to fabricate the Wizard-of-Oz prototype, design some tasks, recruit users, conduct the evaluation of the prototype and report the results. Allocate the role of wizard and the role of facilitator to the relevant staff. Assemble the necessary equipment and inter-connecting software. Select appropriate users to test the prototype, try to cover the range of users within the target population. Prepare realistic task scenarios for the evaluation. Pilot the evaluation procedure and ensure that the wizard is well practised in playing the role of the computer. Check that the recording facilities are available and functioning.

Conduct

The facilitator instructs the user to work through the allocated tasks interacting and responding to the system as appropriate. Any problems or interesting events are recorded by the facilitator and other observers if present. The facilitator then conducts post-session interviews with the users, drawing upon pre-set questions and issues raised during the use of the prototype. After this the users are debriefed and thanked for their co-operation.

Conclusion

The information obtained is then analysed, and the observations and user evaluations are summarised in a report. The major themes and severity ratings characterising the problems are also identified. Design implications and recommendations for improvements can then be fed back to the design team. Again video recordings can support this. Repeating the above process can help to refine the prototype further.

Quality issues

Criteria for successful application of the method

An effective wizard is crucial, so as to ensure the consistency of the users experience and the validity of the obtained information.

Minimum procedure required

The key elements of the process as prescribed above should be carried out. Rating the severity of identified problems is optional although it is recommended in most instances.

Common errors

Most of the difficulties that can arise relate to the role of the wizard and occur because: the wizard fails to act consistently between user sessions or the wizard responds in an unrealistic way that could not be expected of the system in practice.

Further information

Maudsley, D., Greenberg, S. & Mander, R. (1993): Prototyping an intelligent agent through Wizard of Oz. INTERCHI ’93 Conference Proceedings, pp. 277-284.

3.4. Expert-based evaluation

The next group of methods to be discussed are commonly referred to as expert-based or inspection methods. In most cases they involve usability or human factors experts inspecting a system to identify any problems and checking for conformance with style guides as well as established ergonomic principles. The methods discussed here include:

- Heuristic evaluation
- Usability walkthrough
- CELLO - evaluation by inspection.

3.4.1. Heuristic evaluation

What is the method and when can it be used?

Heuristic evaluation, also known as expert evaluation, is a quick technique used to identify general usability problems that operators can be expected to encounter when using a computer or a telematics application. Usually three analysts evaluate the system with reference to established guidelines or principles, noting down their observations and often ranking them in order of severity. The analysts tend to be experts in human factors or HCI, but others, less experienced have also been shown to report valid problems. It is also possible to estimate the proportion of problems found by a group of experts to a theoretical total of problems.

A heuristic or expert evaluation can be conducted at various stages of the development lifecycle, although it is preferable to have already performed some form of context analysis to help the experts focus on the circumstances of actual or intended product usage.

Contribution to design

This method can be used to identify usability problems based on established human factors principles. The method will also provide recommendations for design improvements. This can be achieved quickly but because the method relies on experts the output will naturally emphasise interface functionality and design rather than the properties of the interaction between an actual user and the system.

Deliverables

A list of identified problems, which may be prioritised with regard to severity and/or or safety criticality is produced.

In terms of summative output the number of found problems, the estimated proportion of found problems compared to the theoretical total, and the estimated number of new problems expected to be found by including a specified number of new experts in the evaluation can also be provided.

Benefits

- The method provides quick and relatively cheap feedback to designers and the results can generate good ideas for improving the user interface. The development team will also receive a good estimate of how much the user interface can be improved.
- There is a general acceptance that the design feedback information provided by the method is valid and useful. It can also be obtained early on in the design process, whilst checking conformity to established guidelines helps to promote compatibility with similar systems.
- If experts are available, carrying out a heuristic evaluation is beneficial before actual users are brought in to help with further testing.

Limitations

- The kind of problems found are normally restricted to aspects of the interface that are reasonably easy to demonstrate: use of colours, lay-out and information structuring, consistency of the terminology, consistency of the interaction mechanisms. It is generally agreed that problems found by inspection methods and by performance measures overlap to some degree, although both approaches will find problems not found by the other.
- The underlying statistics (the standard error) for calculating the coverage of found problems compared to "all" problems are not very well understood, and such estimates should therefore only be treated as indications.
- The method can seem overly critical as designers may only get feedback on the problematic aspects of the interface as the method is normally not used for the identification of the "good" aspects.
- It can be very time-consuming to check conformance to voluminous written guidelines so it is often necessary to rely on the expertís knowledge of those guidelines and his/her ability to identify non-conformances ëon-the-flyí. As such the quality of results depends on the capability of the experts who conduct the evaluation.

What you need

People and effort

Two to three human factors experts are required for this method, no end-users are involved. It has been shown that three is the optimum number of experts required to capture most of the usability problems in an interface (c.f. Nielsen and Landauer, 1993). Using larger numbers of experts can be expensive and fail to reveal significantly more problems thus leading to diminishing returns. Under some circumstances a developer may be employed to demonstrate the system although the experts should have hands-on access. Ideally the experts should assess the system independently of one another, although working as a group can also be used. The experts will need approximately half a day each for the assessment (1.5 days total) with between 1 and 2 more days for analysis and reporting. These effort figures will depend upon the complexity of the system being investigated.

Materials and physical resources

All participants should have available the relevant documents, including the standards and heuristics against which conformance is being checked. Sufficient paper and pencils should be provided although it can be more efficient if the notes from the evaluation are typed onto a PC as they are acquired. A system or prototype will also be required for evaluation purposes.

How to apply the method

Preparation

The panel of experts must be established in good time for the evaluation. The material and the equipment for the demonstration should also be in place. All analysts need to have sufficient time to become familiar with the system in question along with the intended task scenarios. They should operate by an agreed set of evaluative criteria.

Conduct

The experts should be aware of any relevant contextual information relating to the intended user group, tasks and usage of the product. A heuristics briefing can be held to ensure agreement on a relevant set of criteria for the evaluation although this might be omitted if the experts are familiar with the method and operate by a known set of criteria. The experts then work with the system preferably using mock tasks and record their observations as a list of problems. If two or more experts are assessing the system, it is best if they do not communicate with one another until the assessment is complete. After the evaluation, the analysts can collate the problem lists and the individual items can be rated for severity and/or safety criticality. Theoretical statistics regarding the number of identified versus unidentified problems can also be calculated if required.

Conclusion

A report detailing the identified problems is written and fed back to the development team. The report should clearly define the ranking scheme used if the problem lists have been prioritised.

Quality issues

Criteria for successful application of the method

Evaluators with little experience will find fewer problems. This might however be compensated for by employing a larger number of evaluators. Evaluators that are both experts in the application area and in human factors issues will prove more effective at detecting potential problems.

Experts that know the rationale and are used to the process and the format of the problem lists, will give better and faster responses. During analysis there is always a need for a judgement call in the decision of whether two stated problems are "the same" or "different". The experts should minimise the risk of misinterpretation when detailing problems. The assessor should also understand a clientís development method and system whilst remaining impartial.

Minimum procedure required

A smaller number of experts can be employed where resources are constrained. There are also statistical methods to determine whether using more experts will be reflected in the identification of additional problems. Furthermore, the severity and/or safety criticality ratings are optional.

Common errors

Hidden assumptions and assessor bias can handicap this method, as can ignoring the context and the tasks. Failing to take into account all the screens or features of the system is another common error.

Finally, if experts work together as a group, and not independently of each other, it is possible that problems identified by individual experts will not be communicated, while there is also no way to calculate all the summary statistics.

Further information

Bias, R.G. and Mayhew, D.J. (Eds.) Cost justifying usability. Academic Press, 1994, pp.251-254.

Nielsen, J. (1992) Finding Usability problems through heuristic evaluation. Proc. ACM CHI'92 (Monterey, CA, 3-7 May), pp. 373-380.

Nielsen, J. & Landauer, T., K. (1993) A Mathematical Model of Finding of Usability Problems. Proc. INTERCHI '93 (Amsterdam NL 24-29 April), pp. 206 - 213.

3.4.2. Usability walkthrough

What is the method and when can it be used?

This is a technique which can be used to identify usability problems in paper-based screen designs, draft training plans, draft documentation etc. Users, developers and usability specialists review a set of designs individually, and then meet to discuss each element of the design in turn. The method is useful early on in the design cycle when a quick and cheap assessment of a system or product is needed, along with an assessment of user reactions.

Contribution to design

The method provides a collaborative forum for users and developers to evaluate different designs and so eliminate defective design elements prior to implementation.

Deliverables

As with a heuristic or expert evaluation a list of problems and their severity ratings is generated and contained within a report.

Benefits

- Detailed user feedback can be obtained quickly and at little expense.
- The feedback can be obtained on paper designs before significant development work is undertaken.
- The walkthrough meeting provides a mechanism to build rapport between users and members of the development team.

Limitations

- Users may be too shy to speak their mind and offer criticisms. This can be a particular problem in the presence of ëexpertsí or people of a more senior position.
- The paper designs which are typically used with this method may not be sufficiently detailed to enable users to appreciate how things will actually work in practice, so the feedback they give must be treated with care.

What you need

People and effort

Up to 5 people may participate in this method, which includes 1 usability specialist with knowledge of HCI issues and 2 people from both the user and designer communities. It is preferable to compose the group so that the users are not outnumbered by the other parties. The usability practitioner will require approximately 2-3 days to arrange and prepare for the meeting, then facilitate the discussion and produce the report. The other participants will require up to half a day each to prepare for and participate in the meeting.

Materials and other physical resources

All participants must be provided with the paper designs which are to be reviewed and a room will be required to hold the walkthrough meeting in.

How to apply the method

Preparation

Agreement should be reached on which designs are to be reviewed. Then those who want to take part should be identified, and a place and time to hold the meeting should be arranged. Provide confirmation of the meeting schedule for all the parties involved and clearly indicate what will be required from everybody. The paper designs should be included with this letter. Each participant must then read through the designs privately, making notes of any points of concern.

Conduct

The facilitator then starts the walkthrough meeting by making sure that everyone is introduced to each other and that everyone understands how the walkthrough will be conducted. Each design element is examined in turn, letting users have their say first in order not to have the specialists dominate the discussion. A list of problems is drawn up by consensus and corresponding severity ratings are defined as they arise. When all the design elements have been looked at, the problem list and severity ratings should be reviewed and any changes that may be required should be made.

Conclusion

At the close of the meeting all the participants should be thanked for their co-operation. A report is then created which documents the identified problems and their severity ratings, it may also include design improvement recommendations. The results of this exercise are then fed back to the development team.

Quality issues

Criteria for successful application of the method

Good facilitation skills are required to successfully manage the walkthrough meeting. Knowledge of human factors, interface design guidelines and standards is also of benefit when evaluating the problem lists and defining severity ratings. Usability analysts that also have in-depth knowledge of the application domain will prove especially effective.

Minimum procedure required

The procedure described above contains all the essential elements.

Common errors

Allowing the developers to dismiss or ridicule the users comments, or allowing the biases of the usability representative to be shown.

Further information

Nielsen, J. (1993): Usability Engineering. Academic Press, pp. 162 - 163.

3.4.3. CELLO - evaluation by inspection

What is the method and when can it be used?

CELLO is largely derived from the expert-based heuristic method promoted by Jacob Neilsen which is widely practised, especially in the United States where the majority of commercial software is developed. The significant characteristic of the CELLO approach is that it is a collaborative technique guided by a defined list of criteria. The criteria may be principles, heuristics or recommendations which define good practice in design and are likely to lead to high quality in use. The criteria represent compiled knowledge derived from psychology and ergonomics theory, experimental results, practical experience and organisational or personal belief.

CELLO can be used throughout the lifecycle but the greatest rewards are to be had by using the method early on in the development cycle. At an early stage CELLO can serve as a check that the user and usability requirements for the system are being observed. In some situations task needs may contravene some general principles, although this is unlikely and can indicate that technological or business requirements are taking precedence over good ergonomic practice.

Contribution to design

Groups of usability experts can use this method to identify usability problems with reference to heuristics and other sources of guidance. The method is more collaborative than conventional heuristic evaluation but also more expensive to apply.

Deliverables

The main deliverable is the evaluation report, which reflects how well a system conforms to a set of principles, heuristics or recommendations that define good practice in design and are likely to lead to high quality in use.

Benefits

- CELLO is a popular method in Scandinavian industry because it is fast, relatively cheap to perform and a simple first step to take on the road to usability testing.
- The method is also easy for organisations to adopt, because it relates strongly to well-understood practices such as the use of design guidelines or principles, and specialist third-party consultancy.
- CELLO as with other inspection methods will always be cheaper than user-based testing and performance analysis.
- During design and implementation, the developing product (from paper prototype to implemented system) can be checked against specific recommendations or guidelines for the particular application or user community.

Limitations

- User trials in context will give more specific information about the defects of a particular application.
- No metric output is provided, only design feedback information.

What you need

People and effort

CELLO is carried out by two or more human factors experts working either alone or in a team. If the participatory nature of CELLO is to be realised to its maximum potential, four to six experts will carry out the inspection. These experts can be both external HCI specialists and internal staff with relevant specialist knowledge. Each analyst will require between 1 and 2 days for preparation, assessment and reporting purposes.

Materials and physical resources

A paper or computer-based prototype is required, and a room to carry out the activity.

How to apply the method

Preparation

An initial context analysis is recommended. Then all the analysts taking part should be identified and the prototype secured. Specific tasks or scenarios of usage should also be obtained. Finally the analysts should agree upon a set of heuristics.

Conduct

Inspection follows a defined procedure which assures the coverage and integrity of the examination. One analyst will chair the meeting. The experts are divided into groups of two, each group having access to a working prototype. Predefined tasks are carried out, problems are noted and then the groups have a structured discussion based around the selected heuristics. The severity of each of the identified problems is then prioritised by consensus.

Conclusion

The evaluation report is then created. This will also explain how specific functions or features of the system contravene the inspection criteria and may provide recommendations as to how the design should be changed in order to meet a criterion or criteria. The results of the inspection are reported in a standard form related to the criteria used and the objectives of the inspection.

Quality issues

Criteria for successful application of the method

Inspection of the finished product against widely-agreed recommendations or guidance, to assure the ergonomic aspects of the system.

Minimum procedure required

The steps outlined above should generally be applied.

Common errors

Permitting individuals to dominate the discussion can prevent the group from reaching a consensus. Many of the problems relating to other expert-based methods also apply.

Further information

Additional details are available from Nigel Claridge of NOMOS in Sweden. Telephone +46 8 7536220, Fax +46 8 7536793, E-mail - nigelc@nomos.se

3.5. User performance evaluation

This next group of methods are closely related to the prototyping techniques discussed previously. They offer the distinctive advantage over expert-based approaches of having an evaluation based on the realistic experiences of actual users. Of the methods presented here there are two broad forms: those that allow a user to interact with a system on their own and those where users and designers collaboratively evaluate the system. Those discussed in this handbook include the following:

- User-based observation for design feedback
- User-based observation for metrics
- Co-operative evaluation
- Supportive evaluation

3.5.1. User-based observation for design feedback

What is the method and when can it be used?

This method offers a relatively quick and cheap way to conduct a user-based evaluation of a working system or prototype. The emphasis is on the acquisition of design feedback and as such fewer participants are required whilst detailed recordings are not essential. The process is linked to the early prototyping methodologies described previously. Observers make notes as users interact with a system to accomplish set tasks. The technique can be used to identify the most significant user-interface problems, but it is not intended to provide reliable metrics.

Contribution to design

This method is aimed at enhancing interface designs by detecting usability problems and providing recommendations to guide re-design efforts. Users work unaided in a naturalistic setting so that the maximum number of problems can be identified, and estimates of usability metrics can be made.

Deliverables

A report which includes a problem list and associated severity levels, as well as recommendations for dealing with the problems is produced. User satisfaction metrics such as SUMI scores are also often included in the report (see the related entry under subjective assessment).

Benefits

- The method promotes the evaluation of a system based on the experiences of actual users, working with the system to accomplish realistic tasks.
- This form of the method is cheaper and quicker to apply than any approach aimed at obtaining metric data.

Limitations

- It is difficult to generalise from results obtained from a few individuals to the entire end user population.
- The method does not yield metric data, so controlled comparative evaluations are not possible.

What you need

People and effort

One to two usability or human factors specialists will be required to prepare, design and conduct the sessions and then to analyse the data and generate the report. It is most cost effective to iterate several times with small numbers of users rather than run one large evaluation. 3-5 users have been shown to be most cost effective (Nielsen, 1993). The application of this method should be achievable in 5-7 days with up to 2 days for the sessions of 1-2 hours, and the remainder divided between planning and reporting the results of the activity.

Equipment and materials

A working version of the product being evaluated is needed. If it is a software product then computer equipment to run it on must also be provided. Any supporting documents (task instructions, questionnaires) must be prepared in advance. Finally a room will be required in which to conduct the evaluation sessions. A usability laboratory or a clients premises could be used.

How to apply the method

Preparation

Note that the output from a usability context analysis meeting can provide the foundation for much of this activity. Enough time should be allowed for any recommendations to be implemented. Identify the people who will take part, the location where the evaluation will be held and the most suitable date. Write to all participants with details of the date, place and what is required of them. Design task scenarios and time limits, write user instructions and prepare all the artefacts and data that will be required for the users to conduct their tasks. Prepare or obtain any necessary questionnaires. Define what information will be collected and how it is to be analysed and presented in the final report - the results of a context analysis can inform this step as can the details contained within a usability plan. Set up the room in which the tests are to take place.

Conduct

Commence the sessions, observing users as they work and making notes as necessary. Only help users if they become completely stuck and there is a risk of missing valuable data.

Conclusion

Collate and analyse the data. Create the problem list and recommendations for design improvements. Summarise the findings and present these in a report for the design team and developers to act upon.

Quality issues

Criteria for successful application of the method

It is important that the design of the evaluation and the task scenarios is carried out with reference to the contextual information that is available. Tasks should be realistic, as should the working conditions if at all possible. A representative sample of users should also be selected.

Minimum procedure required

All the steps outlined above are relevant, the application of the method can be progressed with fewer users although 3 should be considered the minimum.

Common errors

Failing to reassure participants that it is the system that is being evaluated and not them.

Further information

Bias, R.G. and Mayhew, D.J. (1994): Cost justifying usability. Academic press, pp. 245-272. (chapter on ‘Guerrilla HCI: using Discount Usability Engineering to penetrate the intimidation barrierí).

3.5.2. User-based observation for metrics

What is the method and when can it be used?

This form of user-based observation represents a more comprehensive variant of the previous method and entails a detailed analysis of users interacting with the particular system being evaluated. The real world working environment and the product under development are simulated as closely as possible. Observers make notes, timings are taken and video and/or audio recordings made. The observations are subsequently analysed in detail, and appropriate usability metrics are calculated. A prominent example of this approach - NPL's Performance Measurement Method can be used to produce a variety of usability metrics. User-based observation for metrics is particularly suited to evaluating either high-fidelity prototypes or functional systems.

Contribution to design

This method can be used to refine systems by identifying the usability problems associated with the design. The collection of metrics allows a system to be compared against usability targets and/or other systems or versions of the same system. The method also promotes a realistic assessment of the product although it can be more expensive to implement than other approaches.

Deliverables

A report which includes a problem list and associated severity ratings, and recommendations for improvement is generated. Where the performance measurement method has been practised the report will also include one or more of the following usability metrics: effectiveness, efficiency, relative user efficiency and productive period. Again user satisfaction metrics such as SUMI scores are often included in the report.

Benefits

- The effectiveness and efficiency of representative users (or groups of users) carrying out real work tasks in a realistic environment can be quantified and compared with other versions of the product or with target metrics.
- Those features of the product which optimise or reduce usability are identified.

Limitations

- Only real users carrying out representative tasks in a realistic environment (as identified using the Usability Context Analysis method) will provide valid data.
- A minimum of 8 and preferably 10 users are required to obtain reliable usability metrics.

What you need

People and effort

In terms of staffing, 2 analysts are required to implement this approach, one to act as a facilitator and one to control the recording equipment and note any interesting observations. A sample of 8-10 users will also be required although this number could be larger for comparative studies. The users will require up to 2 hours each to participate, as such approximately 3-4 days should be allowed to run the actual evaluation sessions. The analysts will also require this time plus 2-4 days to prepare for the sessions, and between 3 and 5 days to analyse the data and write the report. This suggests a total of specialist effort in the range of 8-13 days although this will depend to some extent on the complexity of the system and the number of metrics which are required.

Materials and physical resources

A computer system or prototype is required for testing, and a room to conduct the study will also be needed. In order to collect the raw data for metrics video recording equipment and analysis tools will also be needed. Video analysis software (such as NPL Usability Services' DRUM package) is commercially available for this purpose. If a usability laboratory is not available portable video and audio recorders will need to be used. Finally all the necessary documents such as task instructions and questionnaires will need to be prepared.

How to apply the method

Preparation

Essential preparation for any user-based evaluation is to undertake a context analysis in order to generate a realistic and valid evaluation plan. One should then identify the people to take part and decide on a location and date for the evaluation to take place. Write to all participants providing them with all the information they require to take part. Task scenarios and instructions should then be created and all materials and equipment needed for the assessment gathered. Scoring procedures with which to produce usability metrics should be determined and recorded in advance. Then set up everything that is required for the start of the evaluation sessions.

Conduct

Run pilot evaluations, including at least one with an expert user if relative user efficiency is required. Run the user sessions and make notes and record what occurs. Following the sessions interviews can be conducted and questionnaires administered to gather subjective impressions from the users based on their experiences with the system.

Conclusion

Analyse the data to produce metrics of effectiveness, efficiency, productive period, relative user efficiency and user satisfaction, as required. Create a ranked problem list and recommendations for design improvements. Compile all these elements into a written report for dissemination to all interested parties.

Quality issues

Criteria for successful application of the method

Ensuring the evaluation closely matches the intended context of use. Use of appropriate tasks. Use of a sample of sufficient size to derive reliable metrics (requires at least 8 people).

Minimum procedure required

All the steps described above should be carried out, although the analysis process can be simplified if only some of the metrics are required and elements such as the productive period and relative user efficiency are not calculated.

Common errors

Failing to control nuisance variables. Using tasks which are not representative. Drawing conclusions based on a sample which is too small.

Further information

A complete description of the Performance Measurement Method is contained in 'The Performance Measurement Handbook', available from the NPL Usability Services, (National Physical Laboratory), Teddington, Middlesex, TW11 OLW.

The topic is also covered on a training course on Performance Measurement supplied by NPL Usability Services (over the course of 2 days).

3.5.3. Co-operative evaluation

What is the method and when can it be used?

This is a cost-effective technique for identifying usability problems in prototype products and processes. The technique encourages design teams and users to collaborate in order to identify usability issues and their solutions. Users work with a prototype as they carry out tasks set by the design team. During this procedure users explain what they are doing by talking or 'thinking-aloud'. An observer records unexpected user behaviour and the user's comments regarding the system. The observer also actively questions the user with respect to their intentions and expectations. This provides qualitative information concerning any difficulties that the users experience and the features or interface elements which give rise to these problems. The method is applicable in the early phases of the design cycle once a prototype is available for evaluation or at a later stage when an existing product is to receive further development.

Contribution to design

A collaborative method which allows users and designers to detect usability problems early in the developed schedule based on the examination of prototypes. It is economical and can be used to reveal problems with small numbers of users, but it is not suitable for the collection of metrics due to the somewhat artificial evaluation conditions.

Deliverables

The primary deliverables from this method consist of a summary of the users' comments and their observed experiences whilst using the system. In addition the method results in a report which lists the identified problems with associated severity ratings and makes recommendations for improvements to the product.

Benefits

- Usability problems can be detected early in the design process.
- Information on the user's thought processes as well as their actions can be obtained.
- Communication between designers and users is promoted.
- The method can be utilised by those with little or no human factors training, and could be practised by members of the development team.

Limitations

- Large quantities of user utterances may be recorded and these can be very time consuming to analyse.
- The close involvement of designers in this evaluation technique makes it unsuitable in circumstances which require an independent assessment, such as quality assurance.

What you need

People and effort

Between 1 and 2 analysts or members of the design team are required for this method, one to act as the facilitator and another to record any observations. If resources are constrained a facilitator can be used alone - as long as adequate recording facilities are in place. In addition one or more users should take part, 5 is the recommended maximum for this type of evaluation as it provides sufficient coverage to identify most of the usability defects. Assuming the participation of five users (for two hours each) the co-operative evaluation sessions could be completed over the course of 2 days. The analyst(s) will also require this time plus another 1-3 days preparation and 2-5 days for analysis and reporting. This suggests a total for the specialists in the range of 5-10 days of effort.

Materials and physical resources

This method requires a functional prototype to evaluate, a location for the evaluation and realistic tasks to occupy the users. Some means of recording the user's comments and actions will also be needed (e.g. audio or video recording, system logs, notebooks). As with other methods all relevant documents such as task instructions, lists of any questions to be asked and/or questionnaires should be prepared. These documents are best pre-assembled into a ready made pack - one pack per user. It is also helpful if each pack contains a blank labelled tape wound to the correct start point.

How to apply the method

Preparation

Recruit users. Identify the target user population, i.e. those who will use the system in the future and select users directly from or akin to this target population. The results of a context meeting can assist in this. Prepare tasks. Select realistic and appropriate tasks which test those features of the system that people will actually use in their work, and those features implemented in the prototype. Write task instructions for the users. Ensure that tasks are clearly expressed, perhaps by conducting a pilot session and working through the tasks and instructions. This will also indicate how much time is required from each user. Prepare any pre-set questions to be asked and clear instructions for those running the evaluation. Ensure the prototype is ready to support the tasks. Ensure recording facilities are available and functioning.

Conduct

Conduct each session, by observing the users as they work through the tasks and recording what they say, and by exploring their impressions and intentions through relevant questions. Make notes of unexpected events and user comments but keep up a dialogue so that the user is encouraged to explain their actions and expectations of the system. Carry out post-session interviews with the users, drawing upon pre-set questions and issues raised during the evaluation. Debrief the users and thank them for their co-operation.

Conclusion

Analyse information obtained, summarise unexpected behaviour and user comments. Consider the themes and severity of the problems identified. Summarise design implications and recommendations for improvements and feed back to the system/process design team. The tape recordings can be used to demonstrate particular themes.

Quality issues

Criteria for successful application of the method

Effective facilitation is required, this is largely based on the skill and experience of the analysts. Suitable tasks given the particular context should be used. A suitable prototype for assessment purposes is also required.

Minimum procedure required

Each of the steps described above are important and should be carried out, the early planning and organisation activities will help to ensure the success of this method.

Common errors

Leading questions and comments from the facilitator which 'put words' into a user's mouth are a problem and should be guarded against.

Further information

Monk, A., Wright, P., Haber, J., and Davenport, L. (1993). Improving your human-computer interface: A practical technique. Prentice Hall International (UK) Ltd.

3.5.4. Supportive evaluation

What is the method and when can it be used?

This is a participatory form of evaluation which can be likened to a 'principled focus group'. The method is used to elicit requirements and re-design information for interactive systems. Users and developers meet together and the user representatives try to use the system to accomplish set tasks. The designers who observe can later explore the issues identified through a facilitated discussion. The method is applicable to a wide range of computer applications and especially custom developments with known stakeholders. It can be applied from requirements elicitation through to pre-delivery checks of documentation and user support. Re-design advice is captured from the users in the structured debrief with a focus on system features and the process can be repeated over a number of trials. One strength of the technique is that it brings users and developers together in a facilitated environment.

Contribution to design

Usability problems are identified via a collaborative forum. The process provides feedback which can be used to advance future system developments.

Deliverables

The output from this method can include: revised requirements, a revised specification document, a report of usability defects (including both diagnostic and summative information) and possibly revisions to the context of use and usability objectives.

Benefits

- The method is reliable and has been established for some years. It is also flexible and easy to report.
- For many projects the application of this kind of method alone can result in quality and training benefits.
- The method promotes a non-confrontational atmosphere and encourages communication between developers and end-users.
- It can be a good way to raise awareness about a product during requirements elicitation and development.

Limitations

- Running the workshop can be hard on the facilitator and so sessions should not be scheduled on consecutive days.
- Only ten system features can be examined per session.

What you need

People and effort

At least 2 analysts are required, one to lead the meeting and another to take note of the issues which arise. At least one of these should have a good knowledge of HCI principles. Ideally the method should be carried out over a number of trials (two are recommended) with intervening periods for re-design activity. Typically up to 4 users per trial take part, for a day. This time is also required by the analysts plus another 8-9 days for preparation, analysis and reporting. Complex trials focusing on systems with many features may require more time.

Materials and physical resources

Test hardware and software will be required for the demonstration and the evaluation phases of the process. Any supporting documentation should also be included. Video and audio recording equipment may also be employed if desired. A room for the meeting to take place in will be required.

How to apply the method

Preparation

Exercises, based on known scenarios of usage, need to be designed. These prompt users to explore the particular features under test in the evaluation while undertaking realistic work. The prototype or system must be stable and all equipment needs to be set up and checked. Typical stakeholders are identified and notified of the arrangements for the meeting. Representative development staff must also be present at the meeting.

Conduct

The meeting commences with an introduction to the project from the developers of the system and to the trial process from the facilitator. Training in the use of the demonstrator, prototype or system can then take place if required. Following this there are hands-on sessions for users who are observed by the evaluators and designers. A de-brief session then follows in which the users and developers discuss the issues under the guidance of the facilitator. Questionnaires can also be used collect further information.

Conclusion

After the initial meeting an analysis of the observations and replies takes place. The findings can be reviewed with the users and then presented to the designers for consideration in re-design plans. The context of use and usability objectives may also need to be reviewed. Further trials can be conducted to assess the impact of design changes.

The intention is for a light-weight technique in which the system developers take away sufficient notes to correct usability defects. However, a detailed report of the findings is often required. This is produced by the collation of notes and flipcharts. If required the notes can be checked against audio and/or video tapes for accuracy. If more than one group of users tests the same system and they come up with the same comments then there is a high degree of certainty that the comments are correct. The method integrates well with an iterative prototyping lifecycle.

Quality issues

Criteria for successful application of the method

The facilitator is the key to this method and he or she should be experienced enough to ensure good results.

Minimum required procedure

Hands on experience for the users plus the facilitated de-brief are the key elements.

Common errors

The method can be sabotaged by bad behaviour by the developers and by poor selection of users. The project should also have resources to implement the findings and these should be implemented before the next round of sessions.

Further information

Support is strongly recommended when first using the method. Experienced facilitators can be provided on a consultancy basis.

Further details regarding the method may be found in the INTUITIVE handbook, available from: Jonathan Earthy, tcsjve@aie.lreg.co.uk, Lloyd's Register House, 29 Wellesley Road, Croydon, CR0 2AJ, UK, tel +44 181 681 4040, FAX +44 681 4839.

3.6. Subjective assessment

The final set of methods to be considered in this handbook are aimed at capturing the subjective impressions formed by users, based on their experiences with a deployed system or new prototype. This can be achieved with the use of questionnaires or through direct communication with the respondents. In the pages which follow four representative approaches are detailed including:

- SUMI
- Cognitive workload
- Focus groups
- Individual interviews

3.6.1. SUMI

What is the method and when can it be used?

SUMI (the Software Usability Measurement Inventory) provides the means to analyse the subjective opinions of users who have practical experience of using a software product. The users fill in a standardised 50-statement psychometric questionnaire and their answers are analysed with the aid of a computer program. SUMI can be used to assess user satisfaction with a high fidelity prototype or with an operational system and can be used in conjunction with many of the other techniques in this volume for investigating such products. However it can only be used to assess software products - it is not suitable for the assessment of hard copy forms etc.

SUMI can be used in two ‘modes’: (a) as a survey instrument, sent by post or electronic mail to users who fill out the questionnaire on the basis of their experiences with the product being evaluated, and (b) as part of a laboratory evaluation procedure. The description in this handbook largely focuses on the use of SUMI as part of a laboratory evaluation procedure, although many of the issues raised also apply to the use of SUMI for survey purposes. The SUMI Handbook contains appropriate procedures for using SUMI as a survey instrument.

Contribution to design

The method provides an indication of the subjective assessment of a product. It also highlights general aspects of design (such as the level of control or helpfulness) which influence the success of a product. The perceived usability of the design can also be compared with previous versions or with similar products based on the SUMI standardisation database.

Deliverables

A report is produced which includes the following SUMI measures: global, perceived efficiency, affect (likeability), control, learnability, helpfulness; and an assessment of how these results compare with results for similar software (deduced from a database of past results). Detailed analysis of how answers to individual questions differed from the norm may also be included. The results of a SUMI analysis can be readily incorporated into the reports from other methods described in this document.

Benefits

- SUMI provides an objective way of assessing user satisfaction which is an important determinant of things such as staff motivation, productivity, turnover and sickness rates.
- Because SUMIís authors have calibrated the results against large numbers of business systems, it is possible to assess how any particular SUMI score stands up against the industry best scores.
- Because SUMI scores are based on a standardised questionnaire, SUMI results can be compared across different systems.

Limitations

- The results SUMI produces are only valid if the sample used is representative of the user population, if the questionnaire has been administered in the same way to all users sampled, and if the results are carefully interpreted. The questionnaire should also be used 'as is', that is, not edited or added to. Some small variation on this restriction is possible, but it should only be attempted in consultation with the developers of the questionnaire, HFRG at University College Cork.
- Personnel with experience of interpreting the results of SUMI outputs will be able to make more detailed diagnoses than inexperienced personnel; HFRG offer a consultancy scheme as part of their SUMI marketing and some organisations (such as NPL Usability Services) offer training in SUMI interpretation.
- Questionnaires can only provide information of a general nature, they do not identify specific problems which can be related to designers.

What you need

People and effort

One person will be required to administer the SUMI questionnaire and between 8-20 users will usually act as respondents. Each user should be allowed 30 minutes to hear what is required and to fill in the questionnaire, although the questionnaire itself can be completed in as little as five minutes. The administrator of the questionnaire will also require between 2-3 days to analyse the SUMI results and write a report. It should be noted however that much of the effort associated with this method can be subsumed under the time allowed to carry out other activities such as a user-based observation where SUMI is used as an integral part of the data collection.

Materials and physical resources

A software product should have been recently used when the SUMI questionnaire is completed. The administrator will need an instruction sheet which gives step by step instructions on how to administer the questionnaire. The instruction sheet should include a standard statement of what is required which is read out to all users before they fill in the questionnaire, this may be based on the standard set of instructions given on page 1 of the questionnaire. The administrator will also require a blank questionnaire for each user to be surveyed. The SUMI package itself, which includes 50 questionnaires, and the analysis software, costs around £400. It is available from the Human Factors Research Group, University College Cork or one of their authorised agents (e-mail hfrg@ucc.ie or see the SUMI web at http://www.ucc.ie/hfrg/sumi for further details).

How to apply the method

Preparation

Conduct a Context Analysis or refer to one that has already been completed, to ensure that the correct types of users are surveyed and that that it is clearly understood which system or part of a system is being surveyed. Identify the people you want to take part, the room where you want to hold the evaluation and the most suitable date. Inform all participants of the date, place and time as well as what is required of them.

Conduct

Ensure that the users have been using the product being assessed as recently as possible, ideally SUMI will be administered at the end of a more extensive evaluation session. Follow the administrators instruction sheet, hand out the questionnaires, read out the standard SUMI statement and answer any questions. Ask users to fill in the questionnaires.

Conclusion

Analyse the questionnaire results using the SUMISCO program, which produces a standard printout of the results. The SUMI Handbook can also be consulted to aid interpretation of the results. Write the report and disseminate to the interested parties.

Quality issues

Criteria for successful application of the method

Ensure you have an adequate user sample for the power of the test and that the manner of recruitment conforms to psychometric standards. If more than one sample of users is called for by the investigation design, ensure that the number of users to be recruited is realistic. Ensure that each user sample is homogenous and has had equivalent amounts of experience with the system and that there are no users in the sample who stand out as being atypical. For instance, avoid including a few users with special (i.e. uncharacteristically high or low) expectations because of their job function or experience (e.g. members of the design team, employees at the organisations head office). If a de-briefing procedure is called for, ensure that all users will receive the same de-briefing. Look over a completed report by someone who has had some experience with interpreting SUMI before writing a report based on SUMI results yourself; better still, make use of the consultancy facilities available with SUMI or attend a training course.

Minimum procedure required

There is a manual provided with the questionnaire: ensure that you are familiar with the contents before trying it out. Unlike software questionnaires will not issue an error message if they are applied in circumstances for which they are not intended. Check that your user sample conforms to the requirements of the target user population as defined in your context of use analysis. The copyright issues associated with the questionnaire must also be respected.

Common errors

Failing to ensure that all the users understand what computer system (and if required, which aspects or parts of the system) is being evaluated, and that their responses should be given in terms of this specific system or parts of it. Failing to check that all the questionnaire items apply to the system being evaluated can also lead to difficulties. Changing items can have serious consequences, and the implications in terms of the resulting loss of reliability, validity, and interpretability must be understood. If the questionnaire has to be translated then the translation used must be sanctioned by HFRG. Users may also fill in the questionnaire unreliably if they have not been briefed about the procedure for filling out the questionnaire.

Further information

The SUMI User Handbook (which is included with the SUMI package) contains detailed information and instructions. See also the SUMI website at http://www.ucc.ie/hfrg/sumi.

3.6.2. Cognitive workload

What is the method and when can it be used?

The measurement of cognitive workload involves assessing how much mental effort a user expends whilst using a product to accomplish a task. This information can be obtained by a number of means such as the subjective workload assessment technique which is based on three rating scale dimensions: time load, mental effort load and psychological stress load. There are also questionnaires for evaluating subjective perceptions of effort developed by NASA and the technical university of Delft. This entry focuses on two particular questionnaires; Delft's Subjective Mental Effort Questionnaire (SMEQ) and NASA's Task Load Index (TLX). SMEQ has one scale which measures the amount of effort people feel that they have invested. TLX has six scales (mental, physical, temporal, performance, effort and frustration) and is designed to measure the individual's perception of what a task has asked of them. It is also possible to collect objective data from heart rate and respiration measurements. All the measurements are based on the users experiences with a prototype or deployed system and the subjective measures can be based on all types of task context.

Contribution to design

The method allows the mental effort associated with a design to be assessed. This provides an indication of whether the mental effort is acceptable, or whether the design needs to be improved.

Deliverables

A set of mental effort values for users performing tasks are produced and included in a report.

Benefits

- Inappropriate levels of cognitive workload can indicate poor performance and health problems resulting from stress and fatigue (under UK Health and Safety law employers have a legal obligation to protect employees' health and well-being, and to pay attention to problems of mental stress).
- The subjective measures questionnaires are very quick to complete, taking less than one minute per person to fill in.
- Where used objective measures cannot consciously be influenced by individuals and they not only provide a measure of mental effort for a task as a whole, but also indicate how it varies throughout the task - helping to pinpoint particular problem events.

Limitations

- As with SUMI the cognitive workload questionnaires depend upon the use of a sample which is representative of the user population.
- As indicated previously questionnaires do not reveal the exact nature of any interface problems which may exist.
- Taking physiological measurements such as heart and respiration rate can itself appear intimidating.

What you need

People and effort

One person will be required to administer the questionnaires, and the users can complete these very quickly (in a matter of minutes). The exact number of users depends upon the nature of the evaluation as these questionnaires are usually administered as part of a wider programme of user-based evaluation, although a range of 8-20 is usual. As such analysis and reporting time (usually 1-4 days) can be factored into the effort required for other activities. If physiological data on cognitive workload are to be collected then 1-2 heart rate monitor technicians plus a heart rate analyst will be required and the data can take a greater amount of time to analyse. As these kinds of measurements depend upon specialist skills it is recommended that the WIT lab at the Technical University of Delft is contacted for further guidance. The collection of subjective data will be focused upon here.

Materials and physical resources

The questionnaires themselves will be required to collect the subjective data on cognitive workload. Copies of the SMEQ and/or TLX questionnaires are available from the WIT Lab. The WIT lab can also provide the equipment to monitor heart and respiration rates. See the section on further information for contact details.

How to apply the method

Preparation

As part of a broader evaluation programme the preparation steps for the user-based observations should be followed prior to using the cognitive workload questionnaires.

Conduct

After each task or subtask of interest is completed the users can be instructed to complete the SMEQ. For the TLX the user is given instructions and can be allowed to practice and become familiar with the questionnaire before they carry out any tasks. The user then completes the TLX ratings and weighting after each task, as with the SMEQ.

Conclusion

The individual answers to both questionnaires are analysed following the instructions provided with the products. A software application is provided to support this process. These results can then either be reported separately or incorporated into the report of other evaluation activities.

Quality issues

Criteria for successful application of the method

The user sample should conform to the requirements of the target population as defined in a related context of use analysis. As with the SUMI questionnaire this sample should be homogenous and have had equivalent amounts of experience with the system. Furthermore the same instructions should be administered to each respondent.

Minimum procedure required

All the steps outlined above should be adhered to as well as any relating to other activities of which the assessment of cognitive workload is a part.

Common errors

Failing to explain what is required of the respondees when completing the questionnaires can be a problem. The users should also be selected with the results of any context analysis in mind.

Further information

Contact Bert Arnold or Marian Wiethoff at WIT Lab, Technische Universiteit, Kaanalweg 2B, postbus 5050, 2600 GB Delft, The Netherlands; Tel +31 15 783720/ 783 753.

3.6.3. Focus groups

What is the method and when can it be used?

A focus group brings together a cross-section of stakeholders in the context of a facilitated but informal discussion group. Views are elicited by the facilitator on topics of relevance to the software product being evaluated. Although focus groups are often used to identify initial requirements they can also serve as a means of collecting feedback once a system has been in use or has been placed on field trials for some time. Several focus groups can be run to concentrate on different topics or to include the views of different sections of the user community. The meetings are usually taped for later analysis.

Contribution to design

This method provides a means of collecting evaluative comments and design feedback from groups of users. Involving users in this way can also promote user acceptance of a new system.

Deliverables

A report detailing the themes and issues raised during the focus group meetings is produced. This report also highlights the design implications raised by the exercise.

Benefits

- The method allows the analyst to rapidly obtain a wide variety of views from a range of people who might have different but equally relevant perspectives.
- Due to the freeform nature of focus groups - unexpected viewpoints may be identified which might have been otherwise overlooked if a more structured approach such as a questionnaire methodology had been taken.
- Focus groups can be more cost effective than individual interviews if a central meeting place is used and the need to travel to the location of various individuals is eliminated.
- Running focus groups can help users accept new technologies - particularly where the organisation or working patterns will need to change as a result of new developments.

Limitations

- Social factors such as peer pressure may lead to inaccurate reports. Although effective chairing can compensate for this.
- Focus groups can produce lots of information which can be difficult to assimilate - in contrast to the subjective feedback obtained by questionnaires.
- Participants in a focus group can have difficulty in articulating their concerns, what they say they do may also differ from what they actually do.

What you need

People and effort

At least one facilitator is required to run a focus group and a second person can help make notes if recording facilities are not available. The facilitator will require training in order to keep the meetings focused. A maximum of 8 participants per focus group is recommended. A focus group may be run in as little as 1-2 hours, although 1-2 days preparation time is likely and a further 2-3 days will be required to analyse the information collected and report the findings.

Materials and physical resources

Meeting space is needed and audio-visual recording facilities should be available if a tape record is required. Access to the system under question can also aid the discussion and permit people to demonstrate particular points. The facilitator should have a list of any important questions that must be asked, and separate sheets of paper will be required for the attendees to record relevant profile information (such as age, occupation, amount of experience with the system etc.).

How to apply the method

Preparation

A facilitator for the focus group should be appointed. Then a range of issues to be addressed is drawn up after which a group of between 6 - 8 representative users is invited to attend the meeting. Each focus group meeting should last an average of 60 minutes. It can also be valuable for the users to have access to the product during the meeting and this should be arranged.

Conduct

The facilitator introduces the issues to be discussed, and clarifies his/her role as an observer and facilitator of free discussion between the users. The facilitator may attempt to ‘draw out’ opinions from users who say very little, and to suggest that users move on to another topic at appropriate points. However the facilitator should avoid too much intervention in the discussion and should not make evaluative comments. The primary task for the facilitator is to listen to and record the discussion. Tape recordings may also be made to aid in later analysis and to present to others not present at the meeting.

Conclusion

Write up the results in a report which clearly identifies the themes and issues. Multiple focus groups are frequently used with the proviso that no user should be present in more than one group in order to get as wide a range of views as possible. If different facilitators are used for some of the groups, then the results can be more convincing still.

Quality issues

Criteria for successful application of the method

Effective chairing of the group is essential as is the selection of appropriate participants.

Minimum procedure required

Fewer focus groups can be run where resources are limited.

Common errors

Leading questions from the facilitator or open displays of surprise, dismay etc. following comments from the participants.

Further information

Blomberg, J., Giacomi, J., Mosher, A. & Swenton-Hall, P. (1993). Ethnographic field methods and their relation to design. In: Schuler, D. & Namioka, A. (Eds.) Participatory Design: Principles & Practices. New Jersey: Lawrence Erlbaum.

Caplan, S (1990) Using Focus Group methodology for ergonomic design. Ergonomics, 33.5, pp.527-537.

3.6.4. Individual interviews

What is the method and when can it be used?

This technique provides a quick and cheap way to obtain the subjective opinions of users based on their practical experience of using a product. Users are interviewed and asked to give their views on the products usability. The interviewer may base his/her questions on a pre-specified list of items or allow the user to freely provide their views, these are referred to as structured and unstructured interviews respectively. Alternatively a combination of these approaches can be practised which results in a semi-structured interview. Semi-structured interviewing is useful in situations where broad issues may be understood, but the range of respondents' reactions to these issues is not known or suspected to be incomplete. Structured interviewing should only be carried out in situations where the respondents' range of replies is already well known and there is a need to gauge the strength of opinion. Unstructured interviews are required where the topic under scrutiny is not understood to any clear extent and is probably the form least likely to be practised alongside a usability evaluation. The other kinds of interviews are useful for obtaining supplementary information after a user has just carried out a usability test. As such they form the basis of the de-briefing sessions which are fundamental to many of the methods discussed previously. They may also be used to gain a broad indication of the usability of an operational product. However interviewing is not a simple task and depends to a great extent on the skills of the interviewer particularly when practising the less structured forms of the method.

Contribution to design

Usability issues and design feedback can be obtained as well as further clarification of usability problems as a result of discussing incidents observed during user-based evaluations. This information can be directed towards producing a system which is more in line with users' perceptions and expectations. Carrying out large numbers of individual interviews can be more costly and time consuming than running one focus group however.

Deliverables

A report summarising the views of those interviewed and providing a number of conclusions based on those views is generated. An appendix containing the detailed write-ups of each interview as well as any questions asked may also be included. As part of a wider evaluation activity the results from interviews can provide supplementary data for the report of a user-based evaluation.

Benefits

- Interviews are quick and relatively cheap to carry out, particularly compared to observational methods.
- Users often value the opportunity to voice their opinions and this can also promote 'buy-in' to the development programme as the users come to feel that their views are being taken account of.
- They can also be useful for identifying areas which require more detailed analysis.

Limitations

- What users say often depends on the skill of the interviewer, particularly when it comes to putting the users at their ease.
- Interviewers may put their own interpretation on what is said. In addition it is possible that interviewees may provide the answers that they believe are expected or that might win them favour. A biased response can be a particular problem if the interviewer is higher up the organisational hierarchy than the interviewee.
- Writing interview notes up can be time consuming. Writing notes based on an audio recording is even more laborious.
- The interviewer may need to acquire domain knowledge in order to know what questions to ask.
- People can have difficulty in articulating their concerns and what they say may differ from what they actually do.

What you need

People and Effort

One interviewer will be required. The number of users to be interviewed will depend upon the circumstances but if the interviews are to be conducted as part of a user-based observation then as a rough guide 3-10 people may be employed. The interviews themselves may last between half an hour and one hour. As a self-contained exercise individual interviews will require 1-5 days of effort for preparation, interviewing, and writing up the results. As with the questionnaire methods much of this effort can be accounted for as part of the overheads of applying the user-based methods described previously.

Materials and physical resources

The interviewees should have had access to the product which is the subject of the interview. Any specific questions that are to be asked should be documented beforehand with spaces left for the interviewer to fill in the answers. A copy of this document should be created for each user being interviewed.

How to apply the method

Preparation

Identify the people you want to take part, the places where you want to hold the interviews and the most suitable dates. Inform all participants of the date, place and what is required of them. Make sure the interviewer is familiar with the product and the organisation in which it is being used. Define any specific questions to be answered and create a sheet for each person to be interviewed.

Conduct

Conduct the interviews taking notes throughout. The interview should commence with a brief 'warm-up' phase to put the interviewee at ease. The interviewer should also take a neutral stance and not ask leading questions. Open-ended questions should be used wherever possible. As a general rule it is best to avoid having more interviewers present than interviewees.

Conclusion

Write up each interview. Identify the common strands running through each interview and identify any particularly important things that were said. Derive some conclusions and write the report.

Quality issues

Criteria for successful application of the method

This method requires considerable preparation on the part of the interviewer. Training in interview techniques would be desirable as would experience of interviewing, especially for unstructured interviews. An analytical but neutral stance is required from the interviewer as well.

Minimum procedure required

Each of the aspects described above should receive attention.

Common errors

Influencing the information by asking leading questions or displaying overt reactions to the responses provided during the interview needs to be guarded against.

Further information

Interviewing is widely covered in the literature, for example see:

Blomberg, J., Giacomi, J., Mosher, A. & Swenton-Hall, P. (1993) Ethnographic field methods and their relation to design. In: Schuler, D. & Namioka, A. (eds.) Participatory Design: Principles & Practices. New Jersey: Lawrence Erlbaum.

Spradley, J. (1979): The ethnographic interview. Holt, Rinehart & Winston, Inc.

4. Selecting an appropriate method

4.1 Selection criteria

There are numerous qualities associated with the different methods discussed in chapter 3, and each of these can have implications for the costs and benefits of using a particular method. In this chapter the various issues of relevance are discussed in order to answer the question - how do you choose which method to use? The decision is aided by the following criteria for cost-benefit analysis, which are included in Table 1:

- the stage in the development lifecycle,
- the kinds of information provided by a particular method,
- resources required: the number of usability analysts, number of users, and the number of person-days required.

The selected methods need to be combined to create a coherent usability programme which effectively deploys user-centred design effort in a particular project.

4.1.1 The stage of the lifecycle when usability input is being considered

Due to the flexibility of many of the methods discussed in this handbook they can be used across a range of systems and development stages. Nevertheless it is possible to provide a general indication of their scope and so refine the choice.

It is generally the case that user-centred design activities can be more cost-effectively implemented in the earlier part of the lifecycle before there is a significant investment in coding. Usability planning and context gathering methods should be carried out during the initial planning and feasibility stage of software development projects, although these methods are relevant whenever a usability assessment of some form is planned. Then during the early stages of the lifecycle when design options are being explored methods relating to guidance and standards, expert-based evaluation and early prototyping are appropriate. Finally in the latter stages of the lifecycle, when implementation and testing are the major concerns, formal user-based testing, performance measurement and subjective evaluation methods are relevant. These distinctions are summarised in the third column of Table 1.

4.1.2 The type of information that is required

The specific outputs from the methods described in this handbook will depend upon the method, the stage of the lifecycle it is applied to, as well as the needs of the project. An indication of the deliverables to expect from the various methods was provided in the last chapter. However, different methods can produce different results which may determine their relevance for a given project. The results obtained from the methods fall into six categories:

- *Plans* - for how to build usability into a system, for which methods should be applied, and how to apply specific methods. These should be integrated with other project development plans and implemented as appropriate.
- *Designs* - what an initial design should look and function like. This information should be embedded within specifications and/or used to create prototypes.
- *Design feedback* - identification of problems and recommendations for improvements to existing prototypes and designs. These should be prioritised and integrated with other bug fix and enhancement requests. Criteria for setting priorities for fixing usability problems can include: the frequency with which the problem occurs for individual users, the number of users who are likely to be affected, the severity of the consequences of the problem, the impact on user satisfaction (implications for motivation, staff turnover, etc.), the cost of fixing the problem and whether other measures such as training could be used to provide a solution.
- *Usability metrics* - numeric values for indicators of usability, such as effectiveness and efficiency. The data can be used to compare: one system with another to demonstrate which will be best, different versions of the same product to identify improvements, different groups of users (or with the same group after different types of training) to demonstrate the need for training or the impact of training, and also to indicate relative levels of user satisfaction.
- *Quality assurance information* - the extent to which the system conforms to stipulated requirements. This kind of information can also be used to demonstrate conformance to established standards (such as style guides) and to ISO 9001 quality standards.

As well as the above types of information data may also be gained on how useful the chosen method was, how many people were required, and what costs were incurred. This information can be used to inform later user-centred design and assessment activities. Based on the six categories discussed above the fourth column of Table 1 indicates the type of results that can be expected from each method.

4.1.3 The resources required

The resources required to put the technique into practice are an obvious factor in determining the costs and benefits. These resources can include personnel and equipment overheads. The method descriptions in chapter 3 included details of the required number of specialist personnel (or analysts), the numbers of end-users and the number of person days of effort. For ease of reference these figures are summarised in Table 1. When selecting a technique one should also check the equipment requirements shown for the method - although for many these are minimal. If more sophisticated recording and analysis tools are not available then some methods may require an initial investment in equipment but these items can be re-used over successive projects and thus the costs amortised.

4.1.4 The inherent benefits and limitations of individual methods.

Human-computer interaction is a young domain and usability assessment methodologies are being continually refined. As such no individual method is perfect, although the ones presented in this handbook have been used successfully in industry. The benefits and limitations associated with each method are described in chapter 3. These cover a range of issues and the reader is referred to those descriptions for specific details. However, some general statements can be offered here to highlight the broad characteristics associated with the different types of method which should be borne in mind when making a selection.

- Planning and context gathering methods

  These methods help to define the scope of usability activities and to ensure that later evaluations are based upon realistic expectations and a sound awareness of contextual issues. The methods are dependent upon the participation and support of other members of the organisation such as developers, project managers as well as end-users. These people, and particularly those in influential positions, must be committed to the principles of user-centred design for later activities to be a success.

- Guidance and standard related methods

  These methods as well as the original sources of documented guidance are likely to become increasingly important with the growing prominence of the relevant standards. Historically the information contained within international standards has proven either difficult to obtain or difficult to apply for many designers. The methods described in this handbook address these concerns. However, the dissemination of advice and guidance based on standards should be sensitively handled and ideally achieved in a co-operative atmosphere to present conformance to the standard as an asset rather than a burdensome obligation.

- Early prototyping methods

Another category of methods that can be used to identify usability problems in the earlier stages. These methods have the advantage of involving end- users and in many cases only need basic materials to create the prototype - although low-fidelity prototypes are not suited to the collection of metric data.

- Expert-based evaluation methods

  These methods provide a cost-effective means to identify known types of usability problem and can be applied from early in the lifecycle. However, they are limited by the skill of the practitioners and cannot be used to identify unpredictable problems which only arise with end-users.

- User-based testing and performance methods

  This category of method also has the advantage that end-users are directly involved in the process and typically make use of the system in question under realistic circumstances and task objectives. This also provides the best setting in which to calculate metrics which are essential for objective comparisons. However methods geared towards metrics are more time consuming and resource intensive to apply than many other methods.

- Subjective assessment methods

  These kinds of methods are typically cheap and quick to apply and are compatible with many other approaches (see section 4.3). However, subjective impressions must be treated with caution as the recruitment and size of the sample is crucial to their validity. There is also the danger that developers can place too much emphasis on subjective data and become complacent - particularly where the users like a system despite the presence of numerous usability problems.

4.1.5 Formal analyses of costs and benefits.

Apart from the pragmatic considerations outlined above there have been attempts to formally analyse the costs and benefits associated with usability assessment activities. A thorough treatment of these is presented by Bias and Mayhew (1994) in their edited book - 'Cost-Justifying Usability', while an illustrative method is provided in the planning section of chapter 3 of this handbook. The typical strategy is to assess the financial savings to be gained by improvements to a system particularly where the result will be increased performance, fewer errors etc. This type of information can be used to justify the existence of a programme of usability assessment. As the calculations will depend upon the exact nature of the system and the organisation's costs for staff etc., it is beyond the scope of this handbook to provide a detailed analysis along these lines. However the principles discussed in the cost-benefit analysis method presented in chapter 3 as well as the resource details provided for all methods will help the reader assess the cost-benefits in these terms.

Another, perhaps more relevant, strategy is to formally assess the relative costs and benefits of different usability assessment methods, where the financial costs of applying the method are weighed against the actual utility of the method for detecting problems with a system. For example various specialists (e.g. see Nielsen, 1993) have put forward a number of observations on the effectiveness of expert and user-based methods in detecting problems. These suggest that:

- expert-based methods, where up to five analysts are employed, are capable of identifying up to 75% of the problems contained within an interface
- expert-based methods will not find all the problems present in a given interface, as the analysts often do not have the same domain expertise as actual users or the same mental models
- expert and user-based methods have been found to highlight different kinds of problems

Expert-based and user-based methods are equally cost-effective in detecting usability problems, and the choice will depend on the life-cycle stage and the availability of experts and users. In general expert-based methods are more convenient early in design, and user-based methods are more important late in design, although both types of methods are potentially valuable at all stages. The combination of the expert- and user-based methods in an iterative programme provides the best chances of problem detection.

On a more general note - the costs and benefits of a given usability method are not static properties, they will depend upon the point in the lifecycle at which the method is applied, as indicated previously. The earlier in the lifecycle the greater the benefits and the lower the costs as it is always more expensive to resolve problems later in the development schedule.

4.2 Using the summary table

Table 1 summarises much of the information discussed above as well as that contained in the individual method descriptions. It lists the methods discussed in this handbook and provides corresponding details on:

- *the applicable stages of development*: based on the planning, early and late categorisation of lifecycle stages. As some methods may be used across different stages the information contained in the third column often shows the most relevant stage for a particular method and then possible additional stages in brackets
- *the types of results provided*: this distinguishes between: plans, designs, feedback, metrics and quality assurance information. Some methods can provide several types of results

- *the number of analysts required*: this reflects the details provided in the individual method descriptions
- *the typical number of analyst days needed to apply the method*: this column provides a range of person days to indicate the likely minimum and maximum required to use a particular method. The number of days that users will be required can be found in the individual method descriptions but will obviously fall within this range
- *the number of users or developers required*: the final column provides this information as either an exact figure or a range for the number of participants.

To help in the selection of an appropriate method, take a copy of this table and then tick those boxes in each column that meet your requirements. The techniques that have the most ticks across the page are most likely to be what is required. To then select between shortlisted techniques study the descriptions in chapter 3. Consultation with experienced practitioners can also help to make a final choice.

4.3 Recommended 'sets' of methods

Although there are many different methods available to practice user-centred design and usability assessment, they are not all competing alternatives. Indeed when combined together different sets of methods can prove very effective at capturing the widest possible range of problems. The methods which support planning such as Usability Context Analysis naturally feed into and inform all the other types of methods. In addition expert-based methods can serve as a useful precursor to prototyping and user-based work as they permit known problems to be eliminated and allow later evaluations to concentrate on the problems revealed by a user interacting with the system. Furthermore, subjective assessment methods can be utilised in conjunction with other methods and are often an integral feature of the evaluation design - particularly for de-briefing sessions. The following section provides more information to aid the reader when selecting and combining

methods.

4.4 Scenarios of usage for the methods

This section considers five usage scenarios for the types of methods presented in this handbook. These scenarios cover a range of possibilities and differ in terms of: the stage of the development lifecycle, the resources required and the origins of the system (i.e. whether the system is based on a commercial off-the-shelf or bespoke application). The scenarios serve to highlight the contributions made by different methods as well as the opportunities for combining the approaches.

4.4.1 Usability throughout the development lifecycle

Scenario background

This scenario (adapted from an early draft of ISO 13407) relates to a large scale project aimed at developing a bespoke system for use by several hundred staff. The system was designed to support tasks which require individual users to work at high performance workstations. Due to the size and importance of this project a long development cycle was planned which provided scope for a varied programme of usability activities. The project management were also sympathetic to the need for usability input. They wished to ensure that training and support costs for the new system were minimised via good human factors design and that the new system benefited users' performance as well as meeting with their approval.

Key activities

A variety of key activities were possible and these were carried out from the early stages of the development lifecycle through to the latter stages of implementation and testing.

- Planning activities: at the outset of the usability programme a facilitated context meeting was held with representatives of the user and management communities. This allowed important contextual details relating to the proposed system to be collected and organisational and usability requirements to be determined. Individual interviews supplemented this process particularly for those people who could not attend the main meeting. All the key parties were given a chance to review the contextual information to ensure its accuracy. Based on these details and the guidance provided in ISO 9241 Part 11 relevant usability objectives were derived.

- Exploration of design options. User involvement during early prototyping was desired and this was achieved by involving users in the creation and evaluation of simple paper prototypes representing different design options. These prototypes were initially constructed with reference to the identified usability objectives and established styleguides. They were subsequently modified in light of the results of initial walkthroughs and supportive evaluation exercises. Where necessary the usability objectives were updated and amended. The output of this phase of the user-centred design programme was a design specification which fed into the creation of a computer-based prototype.

- Usability inspection of the interactive prototype: an expert-based assessment of the computer-based prototype was carried out by three human factors specialists in order to quickly identify usability defects and features of the system which violated best practice in interface design.

This took the form of a heuristic evaluation based on established standards and guidelines. A range of design problems were identified and prioritised, while potential solutions were also noted. This information was reported back to the members of the development team for incorporation in the next iteration of the prototype. Where necessary the in-house styleguide was amended to minimise recurrence of similar problems in future development projects and to ensure that the styleguide reflected the needs of users as well as the principles contained within relevant industry standards.

- User-based observation and subjective assessment: following the expert evaluation of the prototype a new version was created and this was evaluated with a representative sample of users who used the system to work through realistic tasks under realistic conditions. For this evaluation ten user-based observation sessions were carried out with a view to obtaining metric data on the users' effectiveness, efficiency and satisfaction with the system. Video recordings were made and these were analysed to determine relevant performance metrics. A separate clips tape was also made to highlight particular usability issues for management and others not present during the actual evaluation sessions. Subjective data were collected via the SUMI questionnaire and these was supplemented by semi-structured interviews with the users at the end of each session. The output of this phase of the process was a report detailing usability problems and recommendations for re-design of the system prior to deployment.

4.4.2 Usability late in the development lifecycle

Scenario background

In this scenario usability input is called for at a relatively late stage in the lifecycle when a commitment to code has already been made. A computer-based prototype with a high degree of enabled functionality exists and an assessment of the system is required prior to final implementation and release.

Key activities

- Planning activities: a context meeting was held with representative users and managers. This allowed important contextual details relating to the proposed system, the users and their tasks and environment to be gathered. The information was used to highlight the key usability objectives and provide the foundation for an evaluation plan.

- Usability inspection of the interactive prototype: as a precursor to a user-based observation an expert-based assessment of the prototype was carried out by three human factors specialists following the heuristic evaluation paradigm. Design problems and their potential solutions were identified and fed back to the design team. The key aim of this activity was to identify the immediately apparent usability defects and make any necessary design changes prior to involving end-users.

- User-based observation and subjective assessment: following the expert evaluation a user-based observation was carried out with a representative sample of users. Ten participants individually used the prototype system to work through realistic tasks in realistic conditions. Video recordings were made and metric data on the users' effectiveness, efficiency and satisfaction with the system were collected. The SUMI questionnaire and interviews were used to collect the subjective data on the users' satisfaction and personal impressions. The evaluation team produced a report detailing the identified usability problems and made recommendations for re-design of the system prior to deployment.

4.4.3 Usability on a budget: exploring design options

Scenario background

In this scenario a development team wish to involve users in the early stages of design. System coding has not yet started and the team are anxious to verify user requirements at the earliest possible opportunity. Under these circumstances the evaluation of paper prototypes was proposed as a quick and cost-effective way to explore design options and elicit user feedback.

Key activities

- Planning activities: an initial context meeting was held to gather information relating to the intended use of the new system. The details on the users, their tasks and their environment provided valuable insights to guide the evaluation activity. It also provided the basis for the paper designs and the grounds for selecting people to participate in the evaluation of the prototype.

- Early prototyping to explore design options: paper prototypes were created and these were exposed to small numbers of end-users to elicit evaluative feedback. Through the use of small samples and simple prototyping materials it was possible to refine the prototype over a number of iterations. The results were used as the basis for the subsequent computer-based design.

4.4.4 Usability on a budget: collecting design feedback

Scenario background

In this scenario a computer-based prototype is available for usability testing. However, the available resources are limited and development timescales mean that feedback is urgently required. Despite these restrictions there is a commitment to implementing design recommendations - particularly those of a pressing nature. Under these circumstances the following activities were carried out.

Key activities

- Planning activities: a context meeting was held to gather information relating to the intended use of the system. These details concerning the users, their tasks and their environment were used to guide the design and evaluation activity.

- User-based observation and subjective assessment: a user-based observation was run to evaluate the system. Due to the time and resource constraints faced by the project the decision to collect design feedback rather than metrics was made. In this case a sample of 5 representative users were recruited to work with the system on a number of realistic tasks. Video recordings were not made as observers took hand-written notes of problems. Data were also collected via the SUMI questionnaire to provide a general indication of the participants' subjective assessments. In contrast to an expert-based method this approach had the added benefit of collecting information based on the experiences of actual end-users. The design feedback was used to enhance the system by fixing the identified usability problems.

4.4.5 Usability in the context of COTS and bespoke systems

Scenario background

This scenario overlaps to an extent with those discussed previously, however the integration of commercial off-the-shelf (or COTS) software with bespoke software within one system presents particular challenges. In this context usability methods can play a number of roles and it is these general roles which are considered below.

Aiding the initial selection of COTS software

The range of software available to support a given task domain can be large and making the correct selection may be crucial to the overall success of the system. Usability methods can be used to evaluate competing products and thus aid in the procurement process. A range of methods are relevant in this regard. Commercial software may be assessed for conformance against the content of ISO 9241 or against human factors and ergonomic guidelines as a result of expert-based evaluations. The products can be tested by representative users in order to highlight usability defects and performance differences between alternative packages. The output of such exercises will be a recommendation for a particular product with usability provisos noted.

A user-based observation for metrics can be conducted to measure effectiveness, efficiency and satisfaction and thus obtain a broad view of the quality of the system in use. Differences in quality in use will have a direct impact on the costs of training, support, use and maintenance of the system

Where large organisations intend to purchase a system at a considerable price there is usually scope for the supplier to modify the product in order to correct usability defects and thus meet the purchasing organisation's needs more closely. Examples of this 'tailored' approach to purchasing commercially developed software exist in a number of large organisations and government departments.

Evaluating the integration of COTS and bespoke software

Having selected a COTS software component there is often the need for further development work in order to integrate this with software currently in use within the organisation. In this sense the various usability methods presented in chapter 3 and the usage scenarios discussed above continue to apply. However, there are clearly special considerations. Where usability defects are due to inherent problems with the COTS component of the system the possibilities for change may be limited, even if modifications to the bespoke element can improve the overall quality of the interface. In this case all the components of the broader work system and context of use should be re-considered in an attempt to optimise the overall quality in use. For example the usability evaluations may suggest areas where training programmes for operators could be enhanced to overcome the problematic aspects of the system. Additional support may be required, or a faster platform to overcome performance problems. Alternatively the organisational and task requirements may need to be changed to accommodate the impact of the new system.

Method category	Individual methods	Applicable stages of development	Type of results provided	Number of analysts required	Number of analyst days to apply	Number of users / developers required
Planning	**Usability planning**	Planning (+ early and late)	Plans	1	4	1-4
	Usability context analysis	Planning (+ early and late)	Plans	1-2	2-3	2-8
	Cost-benefit analysis	Planning (+ early and late)	Plans	1	5-20	1-2
Guidance and standards	**ISO 9241 applicability**	Early	Plans / designs / QA info	1	3-5	2-8
	ISO 9241 conformance	Early (+ late)	Feedback / QA info	1	5-10	1 +
Early prototyping	**Paper prototyping**	Early	Designs / feedback	2	5-6	2-5
	Video prototyping	Early	Designs / feedback	2	2-3	0
	Computer-based prototyping	Early	Designs / feedback	1-2	see user-based observation	
	Wizard-of-Oz prototyping	Early	Designs / feedback	2	see user-based observation	
Expert-based evaluation	**Heuristic evaluation**	Early (+ late)	Feedback	2-3	3	0
	Usability walkthrough	Early (+ late)	Feedback	1	2-3	4
	CELLO- inspection	Early (+ late)	Feedback	4-6	2	0
User-based testing and performance measurement	**User-based observation (for design feedback)**	Late (+ early)	Feedback	1-2	5-7	3-5
	User-based observation (for metrics)	Late (+ early)	Feedback / metrics	1-2	8-13	8-10
	Co-operative evaluation	Late (+ early)	Feedback	1-2	5-10	1-5
	Supportive evaluation	Late (+ early)	Feedback	2	8-10	4
Subjective assessment	**SUMI**	Late (+ early)	Feedback / Metrics	1	2-5	8-20
	Cognitive workload	Late (+ early)	Feedback / Metrics	1	2-5	8-20
	Focus groups	Late (+ early)	Feedback	1	3-5	6-8
	Individual interviews	Late (+ early)	Feedback	1	1-5	3-10

Table 1. Summary matrix of usability methods

5. Standards and guidelines in HCI

In this chapter a broad range of standards are reviewed, so that those which are most relevance for the reader can be identified. This is followed by a list of other sources of information such as user interface guidelines and styleguides.

Published standards and Draft International Standards (DIS) can be obtained from the British Standards Institution. Committee Drafts (CD, circulated for national vote) and internal Working Drafts (WD) are difficult to obtain. Until this situation changes, copies can be obtained from NPL.

Standards related to human computer interaction fall into two categories:

- process-oriented: these specify procedures and processes to be followed.
- product-oriented: these specify required attributes of the user interface.

Some product oriented standards specify the requirements in terms of performance rather than product attributes. These standards describe the users, tasks, and context of use and assess usability in terms of user performance and satisfaction to be achieved.

5.1 Process oriented standards

Standards of this type can be used to support the following activities:

- specification of overall quality and usability requirements and evaluation against these requirements (ISO/DIS 9241-11 and ISO/IEC CD 14598-1)
- incorporation of usability into a quality system (ISO/DIS 9241-11).

ISO/DIS 13407 (1997) Human-centred design processes for interactive systems

This standard provides guidance on human-centred design activities throughout the life cycle of interactive computer-based systems. It is a tool for those managing design processes and provides guidance on sources of information and standards relevant to the human-centred approach. It describes human-centred design as a multi-disciplinary activity, which incorporates human factors and ergonomics knowledge and techniques with the objective of enhancing effectiveness and efficiency, improving human working conditions, and counteracting possible adverse effects of use on human health, safety and performance. Chapter 1 of this handbook is based on ISO/DIS 13407.

ISO 6385 (1981) Ergonomic principles in the design of work systems

ISO 6385 sets out the ergonomic principles which should be applied to the design of work systems. ISO 13407 is based on these principles and the description of the aims and objectives of ergonomics which are contained in ISO 6385

ISO 9241-1 (1997) Ergonomic requirements for office work with visual display terminals (VDTs) - General Introduction

This part of ISO 9241 introduces the multi-part standard ISO 9241 for the ergonomic requirements for the use of visual display terminals for office tasks and explains some of the basic underlying principles. It provides some guidance on how to use the standard and describes how conformance to parts of ISO 9241 should be reported.

ISO 9241-2 (1993) Guidance on task requirements

This part deals with the design of tasks and jobs involving work with visual display terminals. It provides guidance on how task requirements may be identified and specified within individual organisations and how task requirements can be incorporated into the system design and implementation process

ISO/DIS 9241-11 (1997) Guidance on Usability

This part provides a broad definition of usability:

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

ISO/DIS 9241-11 explains how to identify the information which it is necessary to take into account when specifying or evaluating usability in terms of measures of user performance and satisfaction. Guidance is given on how to describe the context of use of the product (hardware, software or service) and the required measures of usability in an explicit way. It includes an explanation of how the usability of a product can be specified and evaluated as part of a quality system, for example one which conforms to ISO 9001.

It also explains how measures of effectiveness, efficiency and satisfaction can be used to measure how any component of a work system affects the quality of the whole work system in use. The principles are operationalised in the MUSiC Performance Measurement Method (see section 3.5.2 of this handbook for further details).

ISO 10075-1 (1994) Ergonomic principles related to mental work-load - General terms and definitions

This part of ISO 10075 explains the terminology and provides definitions in the area of mental workload.

ISO/IEC DIS 14598-1 (1996) Information Technology - Evaluation of Software Products - General guide

The concept of quality in use has been used in ISO/IEC 14598-1 to distinguish between quality as an inherent characteristic of a software product and the quality which is achieved when a software product is used under stated conditions, that is, a specified context of use. This definition of quality in use is very similar to the definition of usability in ISO/DIS 9241-11. The use of the term quality in use therefore implies that it is necessary to take account of human-centred issues in evaluating software products.

Quality in use: the extent to which an entity satisfies stated and implied needs when used under stated conditions.

5.2 Product oriented standards

These standards can be used in the following ways:

- to specify details of the appearance and behaviour of the user interface
- to provide detailed guidance on the design of user interfaces
- to provide criteria for the evaluation of user interfaces.

However the attributes which a product requires for usability depend on the nature of the user, task and environment. A product has no intrinsic usability, only a capability to be used in a particular context. ISO/DIS 9241-11 can be used to help understand the context in which particular attributes can be required.

In the product-oriented view, usability is seen as one relatively independent contribution to software quality, and is defined in this way in ISO/IEC 9126 (1991) Information technology - Software product evaluation - Quality characteristics and guidelines for their use:

> ‘a set of attributes of software which bear on the effort needed for use and on the individual assessment of such use by a stated or implied set of users’.

Usable products can be designed by incorporating product features and attributes known to benefit users in particular contexts of use. ISO 9241 provides requirements and recommendations relating to the attributes of the hardware, software and environment which contribute to usability, and the ergonomic principles underlying them. Parts 3 to 9 contain hardware design requirements and guidance which can have implications for software. Parts 10 to 17 of ISO 9241 and other standards deal specifically with attributes of the software.

5.2.1 Input devices

ISO DIS 9241-4 (1995) Keyboard requirements

This part specifies the ergonomic design characteristics of an alphanumeric keyboard which may be used comfortably, safely and efficiently to perform office tasks.

ISO/IEC 9995 (1994) Keyboard layouts for text and office systems

An eight part standard which specifies the layout of the alphanumeric, numeric, editing and function sections of the 48-key alphanumeric keyboard, and the keycap symbols to be used for functions.

ISO/IEC 9995-3 (1994) Complementary layouts of the alphanumeric zone of the alphabetic section

Defines how Latin script characters for other European languages should be input when using a standard national implementation of the 48-key keyboard layout.

ISO/IEC 9995-7 (1994) Symbols used to represent functions

Specifies the symbols to be used on keys such as TAB, SHIFT and CONTROL.

ISO/IEC CD 13251 (1996) Collective standard - Graphical symbols for office equipment

Contains 373 symbols appropriate for use on office equipment, collected from other standards.

ISO DIS 14755 (1996) Input methods to enter characters from the repertoire of ISO/IEC 10646 with a keyboard or other input devices

This standard defines how characters from the ISO/IEC 10646 Universal Character Set can be input using their UCS number, or by selecting a character from a screen menu, or by mnemonic keyboard shortcuts.

ISO CD 9241-9 (1996) Requirements for non-keyboard input devices

This part specifies the ergonomic requirements for non-keyboard input devices which may be used in conjunction with a visual display terminal. It covers such devices as the mouse, trackerball and other pointing devices. It also includes a performance test. It does not address voice input.

ISO CD 14754 Common gestures for pen based systems

This standard specifies a set of basic gesture commands for text editing which would enable a user to operate a pen-based system regardless of its country of origin or manufacturer.

ISO 10741 User Interfaces - Cursor controls

Specifies how the cursor should move in text editing and spreadsheet applications.

5.2.2 Screen and environment

ISO 9241-3 (1993) Visual display requirements

This part of ISO 9241 specifies the ergonomic requirements for display screens which ensure that they can be read comfortably, safely and efficiently to perform office tasks. Although it deals specifically with displays used in offices, it is appropriate to specify it for most applications which require general purpose displays to be used in an office-like environment.

ISO CD 13406-1 (1996) Ergonomic requirements for the use of flat panel displays - Introduction

This part explains the ergonomic trade-offs required when flat panel displays are used.

ISO DIS 13406-2 (1997) Ergonomic requirements for the use of flat panel displays - Flat panel ergonomic requirements

This part establishes ergonomic image quality requirements for the design and evaluation of flat panel displays, and specifies methods of determining image quality.

ISO DIS 9241-5 (1995) Workstation layout and postural requirements

This part of ISO 9241 specifies the ergonomic requirements for a Visual Display Terminal workplace which will allow the user to adopt a comfortable and efficient posture.

ISO DIS 9241-6 Environmental requirements

This part specifies the ergonomic requirements for the Visual Display Terminal working environment which will provide the user with comfortable, safe and productive working conditions.

ISO DIS 9241-7 (1996) Display requirements with reflections

This part specifies methods of measurement of glare and reflections from the surface of display screens, including those with surface treatments. It is aimed at display manufacturers who wish to ensure that anti-reflection treatments do not detract from image quality.

ISO DIS 9241-8 Requirements for displayed colours

This part specifies the requirements for multi-colour displays which are largely in addition to the monochrome requirements in Part 3.

ISO 11064 Control room layout

This eight part standard currently under development includes principles of design and layout, displays and controls, environmental requirements and evaluation.

5.2.3 Dialogue design

ISO 9241-10 (1996) Dialogue principles

This part of ISO 9241 deals with general ergonomic principles which apply to the design of dialogues between humans and information systems: suitability for the task, suitability for learning, suitability for individualisation, conformity with user expectations, self descriptiveness, controllability, and error tolerance.

ISO/CD 9241-12 (1996) Presentation of information

This part contains specific recommendations for presenting and representing information on visual displays. It includes guidance on ways of representing complex information using alphanumeric and graphical/symbolic codes, screen layout, and design as well as the use of windows.

ISO/DIS 9241-13: User guidance

This part provides recommendations for the design and evaluation of user guidance attributes of software user interfaces including Prompts, Feedback, Status, On-line Help and Error Management.

ISO/DIS 9241-14: Menu dialogues

This part provides recommendations for the ergonomic design of menus used in user-computer dialogues. The recommendations cover menu structure, navigation, option selection and execution, and menu presentation (by various techniques including windowing, panels, buttons, fields, etc.). Part 14 is intended to be used by both designers and evaluators of menus (however, its focus is primarily towards the designer).

ISO/DIS 9241-15: Command language dialogues

This part provides recommendations for the ergonomic design of command languages used in user-computer dialogues. The recommendations cover command language structure and syntax, command representations, input and output considerations, and feedback and help. Part 15 is intended to be used by both designers and evaluators of command dialogues, but the focus is primarily towards the designer.

ISO/DIS 9241-16: Direct manipulation dialogues

This part provides recommendations for the ergonomic design of direct manipulation dialogues, and includes the manipulation of objects, and the design of metaphors, objects and attributes. It covers those aspects of 'Graphical User Interfaces' which are directly manipulated, and not covered by other parts of ISO 9241. Part 16 is intended to be used by both designers and evaluators of command dialogues, but the focus is primarily towards the designer.

ISO/DIS 9241-17: Form-filling dialogues

This part provides recommendations for the ergonomic design of form filling dialogues. The recommendations cover form structure and output considerations, input considerations, and form navigation. Part 17 is intended to be used by both designers and evaluators of command dialogues, but the focus is primarily towards the designer.

ISO/IEC 10741-1 Dialogue interaction - Cursor control for text editing

This International Standard specifies how the cursor should move on the screen in response to the use of cursor control keys.

5.2.4 Icons

ISO/IEC DIS 11581 Icon symbols and functions

ISO/IEC DIS 11581-1 (1996) Icons - general

This part contains a framework for the development and design of icons, including general requirements and recommendations applicable to all icons.

ISO/IEC DIS 115811-2 (1996) Object icons

This part contains requirements and recommendations for icons that represent functions by association with an object, and that can be moved and opened. It also contains specifications for the function and appearance of 20 icons.

ISO/IEC DIS 115811-3 (1996) Pointers

This part contains requirements and recommendations for icons that are logically connected to a physical input device, and that the user manipulates to interact with other screen elements. It also contains specifications for the function and appearance of 5 pointers.

ISO/IEC 115811-4 (Controls) and ISO/IEC 115811-5 (Tools) are planned.

ISO/IEC WD 115811-6 (1996) Action icons

This part contains requirements and recommendations for icons that represent actions by association with objects that prompt the user to recall the intended actions. It also contains specifications for the function and appearance of 32 action icons for help, search, filing, editing, text appearance, viewing and mailing.

5.2.5 Multimedia

ISO 14915 Multimedia user interface design - Ergonomic requirements for interactive human computer multimedia interfaces

ISO WD 14915-1 (1996) Introduction and framework

This part contains definitions, and explains how ISO 13407 can be applied to multimedia design.

ISO WD 14915-2 (1996) Multimedia control and navigation

This part contains principles relating to consistency, redundancy, mental work load, robustness and explorability. It describes basic controls for audio-visual media, and the structures used to support navigation.

ISO WD 14915-3 (1996) Selection of media and media combinations

This part explains how individual media should be used, and how they can be combined.

5.3 User Interface Guidelines

C. Marlin "Lin" *Brown*. Human-Computer Interface Design Guidelines. Norwood, NJ: Ablex Publishing Corp., 1988. ISBN 0-89391-332-4.

A good source of guidelines for graphical interfaces.

Sidney L. *Smith* & Jane N. *Mosier*. Guidelines for Designing User Interface Software. ESD-TR-86-278. Bedford, MA 01730: The MITRE Corporation, 1986.

This set of guidelines is widely used in military systems, but is based on mid-80s technology with little on graphical user interfaces. Tagged text and PostScript are available via anonymous ftp at: ftp://archive.cis.ohio-state.edu/pub/hci/Guidelines/.

HyperSAM is a hypertext version of the Smith & Mosier guidelines (http://www.dstc.Bond.edu.au/general/ren/hyperSAM.html). The software is currently available in Macintosh HyperCard form. An accompanying paper describes the project and a README details availability.

U.S. Department of Defense. Military Standard: Human Engineering Design Criteria for Military Systems, Equipment and Facilities. MIL-STD-1472D Washington, DC: U.S. Government Printing Office, March 14, 1989.

Section 5.15 of this standard is largely drawn from the MITRE guidelines. A Macintosh HyperCard stack is available via anonymous ftp at: ftp://archive.cis.ohio-state.edu/pub/hci/1472/. It is available on CD-ROM as part of CASHE:PVS.

Goddard Space Flight Centre, Human-Computer Interface Guidelines

This document, prepared for the Software Automation Systems Branch of the Goddard Space Flight Centre, presents user interface guidelines specifically addressing graphic object-oriented interfaces operating in distributed or independent systems environments. The document is available in HTML.

NASA, Goddard Space Flight Center, CHIMES

The CHIMES project uses guidelines to automate the evaluation of user interfaces. The project description is available in HTML.

TRIDENT, Jean Vanderdonckt, University of Namur, Belgium

The TRIDENT project has produced a bibliography entitled Tools for Working with Guidelines. The bibliography containing 1200 references is available at http://www.info.fundp.ac.be/~jvd/guidelines/guidelines_references.html.

5.4 Design Principles

James D. *Foley*, Andries van Dam, Steven K. Feiner & John F. Hughes. Computer Graphics: Principles and Practice (2nd Edition). Reading, MA: Addison-Wesley Publishing Co., 1990. ISBN 0-201-12110-7.

The second edition of this classic contains a few chapters on input and output devices and user interface architecture.

Brenda *Laurel* (Editor). The Art of Human-Computer Interface Design. Reading, MA: Addison-Wesley Publishing Co., 1990.

This is a popular collection of inspiring readings on design.

Aaron *Marcus*. Graphic Design for Electronic Documents and User Interfaces. Reading, MA: Addison-Wesley Publishing Co. (ACM Press), 1992. ISBN 0-201-54363-9; ACM Order number 703900.

This book contains many examples and includes a comparative study of graphical user interfaces on different platforms.

Deborah J. *Mayhew*. Principles and Guidelines in Software User Interface Design. Englewood Cliffs, NJ: Prentice Hall, 1992. ISBN 0-13-721929-6.

This is an excellent practical guide for effective design.

5.5 Styleguides

Apple Computer, Inc. Macintosh Human Interface Guidelines. Reading, MA: Addison-Wesley Publishing Co., 1992. ISBN 0-201-62216-5. There is an interactive animated companion CD-ROM to these Mac guidelines called "Making it Macintosh", Addison-Wesley, 1993. ISBN 0-201-62626-8.

GO Corporation. PenPoint User Interface Design Reference. Reading, MA: Addison-Wesley, 1992. ISBN 0-201-60858-8.

Hewlett-Packard, IBM, *Sunsoft Inc*. & *USL*. Common Desktop Environment: Functional Specification (Preliminary Draft). X/Open Company Ltd., 1993. ISBN 1-85912-001-6. Available as a UNIX compressed PostScript file via anonymous ftp from: XOPEN.CO.UK in: /pub/cdespec1/cde1_ps.Z. (The .Z suffix means that you need the UNIX uncompress program to create the PostScript file.)

IBM. Object-Oriented Interface Design: IBM Common User Access Guidelines. Carmel, Indiana: Que, 1992. ISBN 1-56529-170-0.

James Martin, Kathleen Kavanagh Chapman & Joe Leben. Systems Application Architecture: *Common User Access*. Englewood Cliffs, NJ: Prentice-Hall, 1991. ISBN 0-13-785023-9.

Microsoft Corporation. The GUI Guide: International Terminology for the Windows Interface. Redmond, WA: Microsoft Press, 1993. ISBN 1-55615-538-7.

Microsoft Corporation. The Windows Interface: An Application Design Guide. Redmond, WA: Microsoft Press, 1992. ISBN 1-55615-384-8. Along with the above two publications, Microsoft provides an online visual design guide for its Windows development environments.

Open Software Foundation. OSF/Motif Style Guide. Englewood Cliffs, NJ: Prentice Hall, 1993. ISBN 0-13-643123-2. An online version of this standard is available in the Silicon Graphics IRIS InSight document system.

Sun Microsystems, Inc. OPEN LOOK Graphical User Interface Application Style Guidelines. Reading, Mass.: Addison-Wesley, 1989. ISBN 0-201-52364-7.

Sun Microsystems, Inc. OPEN LOOK Graphical User Interface Functional Specification. Reading, Mass.: Addison-Wesley, 1989. ISBN 0-201-52365-5.

6. References

Bevan, N. and Azuma, M. (1997). Quality in use: Incorporating human factors into the software engineering lifecycle. In: Proceedings of the Third International Symposium and Forum on Software Engineering Standards (ISESS'1997) (in publication).

Bias, G. and Mayhew, D. (Ed's.): Cost-Justifying Usability. Academic Press, 1994.

Blomberg, J., Giacomi, J., Mosher, A. & Swenton-Hall, P. (1993). Ethnographic field methods and their relation to design. In: Schuler, D. & Namioka, A. (Eds.) Participatory Design: Principles & Practices. New Jersey: Lawrence Erlbaum.

Caplan, S (1990) Using Focus Group methodology for ergonomic design. Ergonomics, 33.5, pp.527-537.

Chin, J.P., Diehl, V. and Norman, K.L. (1988). Development of an instrument measuring user satisfaction of the human-computer interface. Proceedings of CHI'88, May 15-19, Washington, pp. 213-218.

Hix, D. and Hartson, H.R. (1993): Developing interfaces: ensuring usability through product and process. John Wiley & Sons, pp. 249-281.

Lim, K.Y. and Long, J. (1994): The MUSE method for usability engineering. Cambridge University Press.

Maudsley, D., Greenberg, S. & Mander, R. (1993): Prototyping an intelligent agent through Wizard of Oz. Interchi '93 Conference Proceedings, pp. 277-284.

Maguire, M. (1997). RESPECT user requirements framework handbook. HUSAT Research Institute, Loughborough University, UK. Also see the RESPECT website at: http://www.npl.co.uk/respect

Monk, A., Wright, P., Haber, J., and Davenport, L. (1993). Improving your human-computer interface: A practical technique. Prentice Hall International (UK) Ltd.

Nielsen, J. (1992) Finding Usability problems through heuristic evaluation. Proc. ACM CHI'92 (Monterey, CA, 3-7 May), pp. 373-380.

Nielsen J (1993) *Usability Engineering*. Academic Press.

Nielsen, J. & Landauer, T., K. (1993) A Mathematical Model of Finding of Usability Problems. Proc. INTERCHI '93 (Amsterdam NL 24-29 April), pp. 206 - 213.

Ravden, S. J. and Johnson, G. I. (1989). Evaluating usability of human-computer interfaces: a practical method. Chichester, England, John Wiley & Sons.

Rettig, M. (1994): Prototyping for tiny fingers. Communications of the ACM, Vol.37, No.4, April 1994, pp. 21 - 27.

Shneiderman, B. (1992). Designing the user interface: Strategies for effective human-computer interaction, 2nd edn, Reading, MA: Addison-Wesley.

Spradley, J. (1979): The ethnographic interview. Holt, Rinehart & Winston, Inc.

Thomas, C. and Bevan, N., Eds. (1995). Usability context analysis guide. Version 4. National Physical Laboratory, Teddington, UK.

Thomas, C. and Curson, I., Eds. (1995). Performance measurement handbook. Version 3. National Physical Laboratory, Teddington, UK.

Vertelney, L. (1989): Using video to prototype user interfaces. SIGCHI Bulletin, Vol. 21 (2), pp. 57-61.