NPL Report DITC 202/92
June 1992

# An Introduction to Usability Evaluation

Miles Macleod
Division of Information Technology and Computing
HCI Group
National Physical Laboratory
Teddington
Middlesex
United Kingdom
TW11 0LW

This work was commissioned by the Information Technology Division of the Department of Trade and Industry, in support of *Usability Now!* the DTI technology transfer programme in Human-Computer Interaction.

NPL Report DITC 202/92



ISSN 0262-5368

National Physical Laboratory
Teddington, Middlesex, UK, TW11 0LW



Approved on behalf of Chief Executive, NPL
by Adrian Marks, Head, Division of Information Technology and Computing

CONTENTS

# 1. INTRODUCTION

## 1.1. ABOUT THIS BOOKLET

Usability matters to everyone touched by Information Technology, and people are increasingly aware of its significance. Usability reflects all those things about a system which enable people to make use of the system effectively, efficiently and with satisfaction, to perform their work tasks. It is affected by factors ranging from detailed design decisions about screen layout to the broadest organisational issues. The importance of usability has grown with the spread of computers into almost every area of our lives, and IT developers increasingly wish to learn more about how to build a high level of usability into systems. Many of the answers are being found in the field of Human-Computer Interaction (HCI), where professionals and researchers with backgrounds in psychology, computer science, ergonomics and other human and social sciences work together. An excellent general introduction to the topic is given in the *Usability Now!* Guide to Usability (Preece et al., 1991).

To succeed in developing systems which meet the needs of the user and provide an acceptable level of usability, we need some means of assessing just how well a system meets those aims: we must be able to evaluate usability. To be reliable and cost-effective, the process of evaluation must use suitable methods. We also need evaluation methods which help us diagnose just what is wrong with a design, to identify specific factors which make it difficult to use, so that the design can be improved. The earlier this type of evaluation can be performed, the cheaper it is to make changes.

The aims of this booklet are to provide the reader with an accessible overview of usability evaluation, expressed in terms which can be readily understood by people without specialist training in Human Factors or HCI, and to show how evaluation methods can be used beneficially. This booklet explores issues underlying evaluation in general, and usability evaluation in particular. It presents selected example methods, describes them in use, and highlights their particular aims and relative merits.

### 1.1.1. Who should use this booklet

This booklet is intended for people involved in the development of Information Technology products, and especially in technical management, who increasingly find that usability evaluation must be considered a significant part of the product life-cycle. It guides the reader towards specific sources of further information, some directly available from the *Usability Now!* programme, others available commercially. It provides a bridge between the introductory material in the *Usability Now!* Guide to Usability, and the numerous methods and tools presented in the *Usability Now!* HCI Tools and Methods Handbook (The HCI Service, 1991).

### 1.1.2. Some questions about usability evaluation

There is more than one way of evaluating usability. In fact, as the *Usability Now!* HCI Tools and Methods Handbook shows, there is a profusion of approaches. They have diverse costs and benefits. Anyone trying to find the most suitable way to evaluate the usability of a specific system will want answers to a range of questions about alternative methods, such as: what are they? what are the benefits? what resources do they require? (or who can do it for us, and how much does it cost?) when can it be done? what does it deliver? how can we exploit the results? where can we find out more?

This booklet will help the reader find answers to at least some of these questions, and seeks to clarify the central issues in what, on first encounter, can be a quite confusing area. Methods have similarities and differences, and answering each of the above questions can lead to a different way of grouping evaluation methods. The *Usability Now!* Guide to Usability divides evaluation methods, on the basis of the methodological approach, into five broad categories, which are followed in this booklet. The *Usability Now!* HCI Tools and Methods Handbook presents the methods according to an equally important factor: the phase of the design cycle in which they can be used. But why do we need methods at all?

### 1.1.3. Why we need to be methodical

When making use of Information Technology – such as a computer system – for work or entertainment, we generally come to some opinion about how well and effectively the system can be used. But on what do we base these intuitive judgements about the usability of a system? Unfortunately, it is all too easy to make informal judgements on incomplete evidence, things we happen to have encountered, rather than things we have methodically sought out. In consequence, we may find that our opinions change – perhaps quite radically – when we wish to use a computer for different tasks, or as we become more skilled, or when we encounter a different computer system which makes the job much simpler to do.

Designers may judge usability, informally and easily, during the process of design. Sometimes, such intuitive assessments of usability are accurate, sometimes they are deceptively inaccurate: they are based on the subjective views of a single person, and may not represent the views of other users. Expert evaluators, by drawing upon their broader knowledge, skills and experience, can aim to achieve informal evaluations which are more sound and fair. But the quality of any informal evaluation is difficult to gauge, and will depend upon the quality of expertise which lies behind it.

It is not easy to produce a usability evaluation which most other people would consider sound and fair, unless some kind of set procedure, or method, is followed. The method must give results which are valid and reliable. To be valid, each evaluation must come up with assessments or measurements which truly relate to whatever it is we want to evaluate. To be reliable, a method must give near enough the same results in identical circumstances. It may still be necessary in many cases to employ an expert to apply the method; but by ensuring that a method is followed, we will end up in a better position to judge the worth of the evaluation itself.

Readers who wish to employ an outside organisation or an individual professional to carry out usability evaluations will find it helpful to consult the *Usability Now!* Directory of HCI Consultants (Design Council, 1991). Those wishing to find out more about available training courses for usability evaluation should consult the *Usability Now!* Directory of HCI Training (J&G Consultants, 1991).

# 2. EVALUATION RATIONALE

## 2.1. BENEFITS OF USABILITY EVALUATION

Usability is the one software quality most constantly apparent to the end-user of a system. When we come to make serious use of a product, **its usability for the tasks we wish to perform** is central in shaping our opinion of its worth. If the product fails to help us to perform those tasks efficiently, effectively and with satisfaction, then we will be likely to form a poor opinion of it. Sadly, the history of information technology is littered with products with poor usability, products which many end-users find awkward, frustrating or quite impossible to use. This incurs all kinds of costs: costs for the user in terms of complex training and slower, lower quality work; costs for the producer in increased demands on software support, and lost future sales.

Such failures can be avoided by evaluating usability, and learning from the results of evaluation. By incorporating appropriate methods for usability evaluation into system development, producers can assess usability at one or more stages of the development process, and equip themselves to improve the quality of their products. Early evaluation can enable direct improvement of designs or prototypes during development; results from later evaluations can be fed into future versions of systems.

We may also wish to evaluate usability for a more specific reason: to certify that a completed system meets pre-determined usability criteria. This approach is being taken up by major companies, including IBM, and will become increasingly significant as standards for usability are established.

The choice of evaluation method depends upon the purpose of that evaluation, and the resources available. In the next section, we consider what information we need to gain from a usability evaluation, if its results are to be useful in improving a design.

## 2.2. EVALUATION TO IMPROVE USABILITY

Evaluation must provide **diagnostic** information, if its results are to contribute directly to the improvement of a design: data which help identify specific areas where improvements are desirable, data which point to those elements in the design which are causing problems for the user, or which are likely to cause problems. It must provide information which can be fed back into the development of the system, or the next iteration of the system. A number of the methods outlined in PART 3 of this booklet have been designed specifically to facilitate the identification of potential sources of usability problems.

An evaluation method, if it is to be an acceptable and useful tool for improving the design of systems, must be practicable. It must be capable of being used in the circumstances in which a system is being developed. In the commercial world of tight schedules and budgets, this means that a method must be demonstrably cost-effective. The required time-scale for achieving results may be very short, and resource allocation may depend upon there being some clear estimation available of the likely benefits of applying the method.

Evaluation methods emerging from academic research have sometimes been criticised for being too complex, cumbersome and costly, and requiring very advanced knowledge to apply. On occasion such criticism has been valid. There are further methods which, at their present state of evolution, show promise in the hands of their inventors, while being commercially unproved. But there is a growing recognition in the research world of the need and demand for relatively simple methods which are directly commercially applicable. Methods are being developed which demonstrably meet that need; for example, the co-operative evaluation method (Wright, Monk and Carey, 1991 - see *Usability Now!* HCI Tools and Methods Handbook, page 99), and the heuristic evaluation of user interfaces by designers. These, and some other promising examples, are outlined in PART 3.

To make usability evaluation economically acceptable in the commercial world may involve compromises, and some deviation from the very rigorous standards usually maintained in academic evaluation work. In general, methods offering quick and inexpensive evaluation are hard-pressed to deliver results which are as comprehensive or demonstrably valid as more meticulous approaches. Yet they can offer real benefits at low cost. They must be judged on what they deliver. A quick and informal evaluation can often give highly useful information, with the caveat that it may be difficult to judge how trustworthy that information is. More rigorous and expensive methods can be shown to be cost-effective if sufficient benefit can be demonstrated from their use. There are also circumstances in which usability can be safety-critical (for example, air traffic control systems), and here there may be ample justification for applying the most thorough available techniques for usability evaluation.

Where the main purpose of evaluation is to provide diagnostic information, the criteria employed when making evaluative judgements are often not made explicit. For example, an expert evaluator, seeking out usability defects in a design, is unlikely to have an explicitly definable view of what constitute acceptable levels for ease of task performance. In the next section, we look at usability evaluation conducted for a different purpose, one which demands that evaluation criteria are made explicit, and perhaps formally expressed.

## 2.3. EVALUATION TO CERTIFY USABILITY

An increasingly important reason for evaluating the usability of a system is to ensure or certify that it meets specific, predetermined usability criteria. Such evaluation will assume real significance for software producers as usability standards are established, and particularly as those standards become incorporated into legislation. Some legislation is in prospect in the near future. A recent European Directive on minimum safety and health requirements for work with display screens (Directive 90 / 270 / EEC) has several implications for software producers, in terms of the usability levels which may be required for their future products. Precise effects upon producers will depend upon the nature of the implementing legislation produced by Britain, and by other European countries to which products are exported. Readers wishing to learn more about this area can find detailed information about Standards and Usability in the *Usability Now!* Directory of HCI Standards (Stewart, 1991). Legislation provides a strong motivation for ensuring that systems achieve specified standards of usability, but there are other good reasons. The adoption of quantitative usability assessment based upon pre-agreed criteria will enable software procurers to incorporate into contracts incentives for meeting specified levels of usability. There is also a prospect of producers gaining a market advantage, by being able to advertise products as possessing a proven level of usability, exceeding the required standards.

The kind of evaluation required for such purposes is often called **usability testing**, where products are evaluated to establish whether they meet pre-determined usability goals, involving specific criteria. This involves evaluation methods which can deliver **quantitative** data, and it requires a relatively formal structuring of the evaluation, with careful attention paid to the choice of representative users, work tasks and context. There has been much interest in this approach over recent years, despite the difficulties which become apparent when trying to establish operationally defined criteria for some aspects of usability. The development and verification of methods for certifying usability is currently an area of active research. A first requirement is for valid and quantifiable indicators of usability, which can be measured reliably, and cost-effectively: these are **usability metrics**. Secondly, minimum levels must be agreed for different metrics, which a system must attain to be acceptable. If they are to be valid, certification measurements must relate to the use of the system by its intended class of users, performing specified tasks which the system is designed to support, in circumstances which adequately replicate its intended context of use. The current focus for European research into usability measurement is the *MUSiC* Project (ESPRIT II Project 5429: Metrics for Usability Standards in Computing).

Evaluation which is aimed solely at measuring and reporting properties of a completed system is sometimes called "summative" evaluation (see page 109 of the *Usability Now!* Guide to Usability). This does not by its nature require the gathering of diagnostic information, of the kind which helps us to identify the causes for any shortcomings. It may be enough to simply produce a set of usability measurements. However, diagnostic information usually has intrinsic value, if only for future reference, and collecting it may incur little extra expense when carrying out a usability evaluation.

## 2.4. WHOSE VIEW OF THE INTERACTION?

We all know that people see things differently, and that judgements made from one viewpoint may not match judgements made from another. This has serious implications for the evaluation of usability. If an evaluation is subjective, we need to know whose view it reflects, before we can judge the worth of that evaluation. For evaluation results to be valid, the evaluation must tell us about the usability of a system **for its intended end-user**. This lies at the very heart of user-centred design. Yet in the process of evaluation, usability can be assessed from many viewpoints other than the end-user's.

### 2.4.1. The Designer's View

Designers, not unnaturally, tend to find their own systems easy to learn and easy to use. They typically have a high level of expertise with a system, and well-developed knowledge of the underlying details of implementation which define precisely the way a system works. Such practised familiarity can make it difficult to anticipate the kinds of problem other users will encounter. In fact, without some training in evaluation methods, it is far from simple for a designer to judge how easily an end-user will be able to perform work tasks. The designer needs to be fully briefed about the expected levels of skill and knowledge possessed by the intended users; about specific work tasks - from all the functions the system supports - which will be typical of the actual work the intended user will be performing, and about the effects of the work setting in which the system will be used.

### 2.4.2. Expert Evaluation

HCI professionals and others who perform expert evaluations (page 111 of the *Usability Now!* Guide to Usability) face some of the same problems. They have the advantage of knowing a lot about what end-users can and cannot do easily. But their own expertise and knowledge will not match the expertise and knowledge of the intended end-user. Experts have to make judgements, predictions and educated guesses about the usability of a system for other people, based on their own use or study of the system. They should take into consideration the anticipated skills and knowledge of the user, the work tasks which the user will be performing, and the intended work context .

There is no reason why the designer should not consider usability in the same way. In fact, some already do. One objective of the *Usability Now!* programme is to encourage more people to consider usability from the viewpoint of the end-user. Generally, some training or guidance in Human Factors is advantageous, when a non-specialist seeks to produce this kind of evaluation. Recent work suggests that a small amount of training can be very cost-effective in enabling non-specialists to perform evaluations.

### 2.4.3. Observational Methods: A More Objective View

So far, we have considered designers and experts evaluating usability without actually involving the intended end-user of the system. If a system or prototype is available, then there is no substitute for getting real, representative end-users to try it out, and observing or measuring how well they get on. This can give a much more accurate picture of usability. It allows the evaluator to measure or assess how usable the system actually is, in the hands of the people for whom it is designed. At their simplest, such evaluations should involve observing intended users - or people with similar levels of training, experience and knowledge of the system and tasks - performing sample tasks with a system, a prototype, or a functioning mock-up. (See page 112 of the *Usability Now!* Guide to Usability.) The circumstances in which the observation is carried out should match as closely as possible the intended normal working environment. **Qualitative** evaluation of performance will involve judgements about what is done easily or with difficulty, and attempted diagnosis of usability defects and their causes. The evaluation can also be **quantitative**, involving timing and measurement of task performance.

A more precise record of the way in which the end-user interacts with the system while performing work tasks can be captured on video. This can be done in a usability laboratory, or in an actual work setting; in either case it is important that the user should find the process of recording inconspicuous, and that it does not interfere with their work. Analysis of a detailed video recording of task performance can provide considerable insight into how well, and how easily, users perform specific operations and work tasks when using the system. It also allows the derivation of quantitative information about the usability of the system.

### 2.4.4. Survey Methods: The User's View

What people think about a system when they use it is a primary factor in determining its usability. Surveying the subjective opinions of end-users can be used as an evaluation technique in its own right, or can provide rich additional information for observational, performance based evaluations. The *Usability Now!* Guide to Usability (p 115) introduces the fundamentals of survey methods. There are some caveats about the manner of collecting subjective information, particularly verbal data. If users are asked to describe what they are doing while they do it, this may change the way they perform their tasks. If they are asked to report some time afterwards, they may miss, or misrepresent, important factors. Methodical collection and interpretation of subjective data allows the evaluator to build up a composite picture of the system's usability from the end-user's viewpoint. Different techniques for eliciting the views of users are outlined in PART 3 of this booklet.

## 2.5. WHEN CAN WE EVALUATE? WHAT CAN WE LEARN?

These two questions are closely related. Usability evaluation in some form can be performed at all stages of system development, from early specification to the final, delivered end-product. What can be learned from any evaluation is affected by the stage of development at which the evaluation is conducted. The descriptions of evaluation methods presented in PART 3 include information about the stage(s) of development at which different methods can be applied. First, we shall consider some general principles.

Early in development – when there is only a specification, and no actual product available to evaluate – usability evaluation relies upon prediction of how people will be able to use a system built to that specification. That prediction must either be derived methodically from analytic models of how people and systems interact, or, as we saw in the previous section, drawn from individual knowledge and experience. In either case, its accuracy depends upon the quality of our underlying models and knowledge, as well as the manner in which they are applied.

When a mock-up or prototype is available for use, user-based testing enables us to evaluate how effectively and efficiently people can make use of it in performing tasks, and with what level of satisfaction. At this stage we can learn considerably more than earlier in development, using observational and survey methods. There is great value in being able to base our judgements on the performance of people, rather than on purely theoretical or speculative predictions. System developers who do not employ prototyping as a regular part of the development process may view the effort required to produce a working mock-up or prototype as a major disadvantage; but there are substantial benefits in evaluating usability and identifying usability defects before too many design decisions have become frozen.

It is worth noting that user-based evaluation still relies upon prediction, in that we have to generalise from the results of a study involving specific users performing specific tasks in specific circumstances, and predict usability for the final population of end-users, carrying out real work tasks in a variety of working environments. The validity of our conclusions relies upon how well a mock-up or prototype represents the complete system, how well the testing represents the work context in which the system will be used, and how well the sample users represent the intended end-users.

Once a system is implemented, user-based evaluation can draw on a pool of real users, and we can learn most about system usability; information which can be exploited in the development of future versions and systems.

## 3. METHODS FOR EVALUATING USABILITY

### 3.1. AIMS AND ORGANISATION

The aim of the following section is to provide the reader with an overview of the different approaches which can be taken to evaluating usability. It is intended primarily for readers without a specialist training in HCI, Ergonomics or Human Factors. It gives overviews and examples of evaluation methods from each of the five approaches to usability evaluation described in the *Usability Now!* Guide to Usability. The aim here is to equip the reader to make informed choices about evaluation, by presenting a clear picture of the similarities and differences between possible approaches, highlighting key attributes, and giving examples of methods currently in use.

For readers who would like greater technical detail about specific methods, references are given to the *Usability Now!* HCI Tools and Methods Handbook and other relevant sources. The FURTHER READING section provides interested readers with sources of technical information about broader issues and specific evaluation methods.

### 3.2. HOW TO CHOOSE AN EVALUATION METHOD

The choice of method for a specific evaluation depends upon the kind of results required, the resources available (funding, expertise and time), and the stage of system development. The initial costs will typically be greater if you wish to be accurate and thorough, although the outlay may be more than recouped in the longer term.

The cheapest and quickest method will often be to employ someone with expertise in usability to produce a specialist report (an EXPERT method). This can give a lot of information very quickly, can diagnose significant faults, and provide specific recommendations. However, much depends upon the quality of the expertise. There can also be difficulties feeding the results back into the design.

Where the priority is to perform usability evaluation early in design, an ANALYTIC method, although narrow in scope, can give very early feedback.

For a more broadly revealing picture of usability, it is advisable to evaluate with real end-users, performing representative work tasks, in a real or simulated work setting. That is, to employ user-based methods. Objective usability data can be gained by observing users, and analysing how well they perform their work tasks (using an OBSERVATIONAL method). This can give both diagnostic and quantitative information about usability. For thorough observational evaluation, it may be best to use the services of a professional usability centre or laboratory, although it is possible to achieve worthwhile results using such methods in-house. For the most effective communication of results back to the design team, designers should have an active involvement in the evaluation process, and see for themselves.

Where users are prepared to participate actively in an OBSERVATIONAL evaluation, and to explain why they do things and what is difficult, then co-operative evaluation offers significant advantages. If staff are particularly interested in usability, it is possible to conduct co-operative evaluations in-house, even without a resident Human Factors professional, but professional guidance and extra facilities may be a help.

The cheapest way to get a lot of information directly from users is to use a questionnaire (a SURVEY method). Preferably use a questionnaire which has been scientifically developed specifically to evaluate usability. Such questionnaires should give trustworthy quantitative data about different aspects of usability, when completed by ten or more users, provided they are used as directed. Questionnaires can also be tailored to elicit diagnostic information about a particular system; in such cases it requires some expertise to interpret the results.

For a comprehensive evaluation of usability, it is sensible to use more than one evaluation method. For example, an OBSERVATIONAL method to evaluate user performance, and a SURVEY method to evaluate how satisfied the users are; or an EXPERT method for quick diagnosis of specific usability problems in a prototype, combined with iterative use of a SURVEY method to test whether changes are successful in improving usability.

Formal CONTROLLED EXPERIMENTAL STUDIES in this area face fundamental problems. They tend to be complex and costly. More significantly, there are so many potentially relevant variables in the characteristics of the users, the tasks they perform, and in the technical, physical and organisational environments in which the system is to be used, that there are considerable difficulties in generalising from the findings of such a study. However, this does not invalidate the usability testing approach. Useful and valid results can be obtained, for example from combination of a carefully structured quantitative OBSERVATIONAL evaluation and a trustworthy quantitative SURVEY method; to enable interpretation of the results, it is essential that the key characteristics of the users, tasks and setting employed in the study are clearly reported.

The Evaluation Methods at a Glance table presents the key points of each approach, in a simple format, and summarises their main advantages and disadvantages. More details are to be found under the relevant headings in the following pages.

## 3.3. EVALUATION METHODS AT A GLANCE

| TYPE OF METHOD | ANALYTIC | EXPERT | OBSERVATIONAL | SURVEY | CONTROLLED EXPERIMENT |
|---|---|---|---|---|---|
| Timing of use (in product development) | early | any | prototype or late | prototype or late | any |
| User-based | no | no | yes | yes | yes |
| Scope of evaluation | narrow | broad | broad | broad | very narrow |
| Type of data: | | | | | |
| • quantitative | yes | no | yes | yes | yes |
| • qualitative | no | yes | yes | yes | yes |
| • diagnostic | narrowly | yes | yes | yes | narrowly |
| Expertise required | high | high | medium | medium | high |
| DIY (in-house) | no | can try... | yes | yes | no |
| Commercial service | yes | yes | yes | yes | yes |
| COSTS: | | | | | |
| • time | medium | low | medium | low/medium | high |
| • money | low/medium | low | medium | low/medium | high |
| MAIN ADVANTAGES | Used early in design<br>Fairly low cost | Low cost<br>Quick<br>Diagnostic<br>Broad scope | User-based<br>Diagnostic<br>Broad scope | Fairly quick<br>User-based<br>Users' views<br>Diagnostic<br>Broad scope | User-based<br>Rigorous |
| MAIN DISADVANTAGES | Narrow scope<br>Not user-based | Not user-based<br>Questionable reliability | Less effective early in the design cycle | Less effective early in the design cycle | Narrow scope<br>High cost |

In the following sections, examples of the five different types of evaluation method are presented, starting with those which can be applied early in the design cycle. This order is intended to ensure that readers can easily relate the material here to the *Usability Now!* Guide to Usability (pages 108-121). However, the following sections need not be read sequentially. The reader may for example wish to explore the more accessible approaches, such as expert and observational methods, before attempting to understand analytic methods.

## 3.4. ANALYTIC EVALUATION METHODS

These methods deliver predictions, usually about quite narrow aspects of usability. They require at least a partial specification of the system and its user interface, and employ some kind of theoretical representations of the user. Theoretical analytic evaluations of usability offer some appealing advantages for system designers. Being based on specifications, they can be conducted early in the design cycle. Results tend to focus on quite detailed aspects of the design, such as menu design and specific dialogue sequences. They can include specific recommendations for change. For analytic evaluations, it is necessary to hire the services of a specialist in analytic evaluation.

Evaluating usability before any code is written is a very appealing prospect, offering considerable potential cost advantages. The results of the evaluation can be fed back into the design process before implementation commences. This is a promise which researchers are currently working to meet, with a degree of success. One pre-requisite for successful theoretical analytic evaluation, which makes it complex, is that it must employ some kind of model of the user: of how easily, or how well, the user can meet different demands which interaction may make. (Readers not convinced that a model of the user is necessary, might consider the following: an analytic method which neglects to model the user's physical and mental capabilities will deliver the same prediction of usability for a system whether that system is being used by an expert, a novice, or for that matter a chimpanzee.)

For the purpose of telling us something useful about an interface design, a model does not have to be highly sophisticated, or represent precisely the way we think. What it must do is provide valid predictions of some aspect of interaction affecting usability. At its simplest, this can merely be the time taken to perform a task, if no errors are made. More useful is some prediction of how difficult the user will find it to perform a task, and the identification of places in interaction where errors are likely to occur. In the next few paragraphs, we shall look briefly at what different types of analytic model can offer, and how an analytic approach is being taken up by European industry.

Some analytic models do not attempt to model directly the cognitive processes of the user. Instead, they model tasks which the user will perform when interacting with the system. They lead to predictions about usability on the basis of the way various properties of tasks influence the way those tasks can be performed. Readers familiar with the *Usability Now!* Guide to Usability (page 110), will have encountered the difference between single layer models, and multi-layer models of interaction. One single layer model, the Keystroke-Level Model, is described below. It is simple to use but very limited in scope. Multi-layer models analyse interaction at several levels of detail, and can provide more sophisticated predictions about the relative difficulty of performing tasks with a system. (See pages 27 & 29 of the *Usability Now!* HCI Tools and Methods Handbook for details of two such methods: CLG and TAG). A major criticism of such methods is that they require very considerable expertise to perform the necessary analysis. Also, many task-analytic methods do not pay sufficient attention to the nature of the output to the user from the system. Visible and audible output plays an essential role informing the user about the state of interaction, and about the kinds of input which are appropriate, hence it is important to evaluate the ease with which output can be understood.

### 3.4.1. The Keystroke-Level Model

The Keystroke-Level Model (K-LM: Card, Moran and Newell, 1980) is one of the oldest and simplest methods for theoretical evaluation of how easy it is to perform a task via a given computer interface. The K-LM is explicitly limited to predicting time for expert, error-free task performance. It is a single layer model, developed from a series of empirical studies and theoretical analyses of how people use text editors. As its name suggests, a K-LM analysis describes interaction at the level of individual keystrokes and mouse movements: the input which the user has to make into the system in order to perform the task. Its model of the user also makes a simple allowance for the time which an expert user, fully practised in using the system, will have to devote to thinking, and to moving hands from keyboard to mouse. By adding together predicted times for individual keystroke-level actions, the K-LM provides a means of predicting the time it will take expert users to perform individual tasks if they do not make any mistakes.

• A typical example of use: 'back of envelope' calculations, made during design, of the times for carrying out frequently used sequences of user actions, which various design alternatives will incur.

The Keystroke-Level Model:
• is quite simplistic
• is applicable early in system development
• has limited goals and very narrow scope
• achieves its limited goals effectively and easily
• is one of the few analytic methods which has gained widespread use
• ignores task context, as well as the wider context in which the system will be used

### 3.4.2. Analytic Process Models

Analytic models can be employed, early in design, to test specifications for specific aspects of usability. Their aim is to reduce the need for design changes later in development, when alterations are more difficult and expensive to make. Process models represent the cognitive processes of the user and the processes of system. They work by generating predictions of the complexity of thought and action which are demanded of the user in performing specified tasks. This is not a simple exercise. It requires a cognitive process model of the user, and a specification of the system user interface. Currently available methods require substantial expertise to deliver useful predictions, although much effort is being devoted to making such methods easier to apply. Kieras (1988) reports a simplified application of Cognitive Complexity Theory. Bösser and Melchior (1990) describe SANE, which is supported by a computer-based toolkit, offering the promise of integration with tools which use or generate formal specifications (CASE and prototyping tools).

SANE is based on the Skill Acquisition NEtwork, a cognitive model of procedural knowledge. It requires a functional specification of the hardware / software system, and models of tasks which are to be performed on it. From these, the SANE toolkit can be used to generate verification of the functional design (confirmation that the prescribed tasks can be performed with the system), as well as an assessment of the efficiency of the work performed with the system, and an evaluation of the extent of knowledge required from the end-user. It gives results which allow the comparison of alternative designs, and which help identify shortcomings such as inappropriate displays, or commands and menu designs which are likely to cause problems for the user. As with other analytic, theory-based evaluations, SANE is restricted to those application areas where formal representations of system components exist. Thus, it is best seen as augmenting rather than replacing user-based evaluation.

• Some examples of use: the SANE toolkit has been used to evaluate a complex workstation in the testbed for the European Space Agency Columbus Project, and to evaluate interactive devices to be used by drivers of road vehicles, which are part of a communication system for fleet management.

Analytic Process Models:
• are applicable early in system development
• employ theoretical models of the user
• are restricted in scope, so need to be augmented by user-based evaluation
• can offer integration with CASE and prototyping tools, when supported by a computer-based toolkit
• require formal specifications of the user interface
• require an expert to apply the method
• do not consider the wider context in which the system will be used

## 3.5. EXPERT EVALUATION

Expert evaluation methods, rather than analysing specifications or studying intended users of a system, draw upon *expert knowledge* to make judgements and predictions about the usability of the system for those end-users. Typically, expert methods involve the expert making use of a system or prototype. Alternatively, experts may report on the anticipated usability of a design from its specification, or from early mock-ups. A prerequisite for an expert report is a specialist with the right expertise. If there is no in-house HCI professional, then it is generally a good idea to employ reputable consultants, with the right expertise, and who are capable of communicating their findings successfully to the client, and also to the design team where appropriate. The *Usability Now!* Directory of HCI Consultants gives details of organisations and individual professionals who can offer suitable evaluation services.

### 3.5.1. Specialist Reports

Specialist reports are a long established, loosely defined way of evaluating the usability of a system. A Human Factors expert (or more than one expert) provides critical evaluation, based upon expert knowledge. Sometimes the expert's knowledge is augmented by specific handbooks and guidelines giving preferred features and style of interaction. For their validity, specialist reports rely heavily upon the quality of the expertise, and the skill of the expert in imagining how the system will match the abilities and preferences of the intended end-users. The expert may be able to provide some useful evaluation from quite early in development, given a simple specification of interaction and a graphical representation of the visible interface.

Where high quality Human Factors expertise is available, this can a very effective form of evaluation. The expert can also suggest potential solutions to the usability defects in the system. The client can usually make a reasonable judgement of the quality of recommendations made in a specialist report. What is much more difficult, though, is to discern what the specialist may have overlooked.

• A typical example of use: a specialist report can be produced about almost any system, during or after development. In general terms, the essence of a specialist report is a list of usability defects and potential problems, which the specialist anticipates intended users will experience. This should be accompanied by a matching list of suggestions about how to prevent (or get around) these problems.

### 3.5.2. Expert Walkthrough

This is a variation of the specialist report, but employing more of a method. The critical evaluation is generated by the Human Factors expert on the basis of 'walking through' a number of tasks, which should be representative of the tasks which the system is designed to support. Well chosen sample tasks make it more likely that the expert will encounter things which the end-user will find important in influencing the usability of the system.

However, expert walkthrough again relies upon the talents of the expert for anticipating which things users will find easy or difficult in using the system.

Specialist Reports / Expert Walkthrough:
- can be used from the early stages of the design process
- are not user-based
- are widely used
- can be relatively cheap and quick
- can diagnose a wide range of problems
- rely on the opinion and judgement for prediction of usability for end-users
- depend upon the quality of the expert(s)
- are less useful for systems or interfaces which break new ground (and hence are outside the scope of existing expertise)

### 3.5.3. Cognitive Walkthrough

Cognitive walkthrough is a recently-developed method which gives expert evaluation of usability (before implementation) a more rigorous theoretical basis. It is in some ways a cross between an expert and an analytic method. It provides an assessment of aspects of usability, and assigns causes to specific anticipated usability problems. Cognitive walkthrough requires task scenarios – low-level descriptions of sample tasks which are going to be performed by users of the system being evaluated – together with a list of the actions that have to be taken by the user to achieve each task via the specific computer interface.

Cognitive walkthrough is based on a model of exploratory learning (CE+). The cognitive walkthrough procedure, assisted by a computer-based tool, leads to an analysis of the cognitive requirements imposed by the interface design. It is reported that analysts are able to use the tool with about an hour's training (for more details see Rieman et al. 1991). Lewis et al. (1990) suggest that use of the cognitive walkthrough method can detect, with a very modest investment in resources, almost 50% of the usability problems which will be encountered by users of a system. However, Jeffries et al (1991) found that some serious problems tend to be missed.

- An example of use: Cognitive walkthrough was used to compare the usability of different versions of a simple email system, at the specification stage. Several small tasks were analysed. The analysis predicted about half the problems which were observed when the implemented systems were used.

Cognitive walkthrough:
- is intended for use early in system development
- is theory-based, but requires some expert judgement
- aims to consider higher cognitive factors
- is inexpensive, and is being made easier to apply
- does not consider the context in which the system will be used
- seems promising, but is yet to be widely adopted

### 3.5.4. Checklists

People often ask if there is some kind of checklist they can use to evaluate usability. What they have in mind is a clearly laid out list which they can work through point by point, to produce a simple but thorough evaluation. Checklists for evaluating usability are available (see p 122 of the *Usability Now!* HCI Tools and Methods Handbook; and Ravden and Johnson, 1989). They are usually in the form of sets of questions an evaluator can ask about a system and its user interface. They offer the advantage of being straightforward to use, and they provide a structure around which to shape an expert evaluation. However, to produce an effective evaluation, checklists do require some degree of expert knowledge, both to answer the questions and to interpret the results.

The aim of a checklist is usually to test how well a system complies with a set of guidelines, or with an underlying set of principles. Such checklist-based evaluation can, at best, only be as good as the guidelines and principles it checks. It further relies upon the evaluator understanding the implications of the various items in the checklist: this poses a problem for user-based checklists, which can easily ask questions beyond the comprehension of some users. There are also difficulties in interpreting the results from completed checklists, although the answers may point clearly to specific problems.

Ravden and Johnson (1989) provide a step by step guide to carrying out a checklist-based evaluation, where the evaluators work through carefully chosen tasks on the system they are evaluating. They give some very sound advice on how to select representative tasks, including consulting and observing real end-users. They suggest involving the designers and software engineers in the evaluation process, which eases the problem of convincing the design team of the worth of the evaluation results.

• An example of checklist use: Ravden and Johnson's method was used to evaluate a project management information package developed in-house by a company, but in need of improvement before wider implementation. The checklist-based method was employed by representative end-users (making this expert method very user-based!). The completed checklists revealed general agreement about what was wrong with the system. Collation of specific answers clarified what and where the problems were.

Checklists:
- can provide guidance for evaluation at most stages of system development
- are sometimes user-based
- are relatively straightforward for Human Factors experts to apply
- are commercially available and inexpensive
- give results which may not be easy to interpret

### 3.5.5. Guidelines and Principles

Guidelines are used in ad hoc evaluations, as well as in checklist methods, and in the design of systems conforming with specific platform styles. One of the first and most influential style guides was the Apple Human Interface Guidelines (Apple, 1987). This sets out in clear and simple terms how to make sure that the design of the user interface of Macintosh applications conforms with user expectations, and explains the underlying principles which influence usability. Many other manufacturers now produce their own style guides.

But guidelines themselves tend to proliferate. Smith and Mosier, working in the USA, have produced perhaps the best known set of general guidelines for the design of user interface software, comprising about one thousand design guidelines (see p 53 of the *Usability Now!* Guide to Usability; p 125 of the *Usability Now!* HCI Tools and Methods Handbook). Individual guidelines may lead the designer in different directions (e.g. "present all the necessary information on screen" versus "keep the screen uncluttered"). Hence applying a set of guidelines means making compromises and trade-offs. It can be difficult to evaluate the quality of those trade-offs. Guidelines do tend to be system or platform specific, and based on assumptions about the kind of interface. Recent work shows that interactive examples are more effective than written guidelines in helping designers to achieve a standard house style with good usability.

Usability principles are much more general statements about things which affect usability (such as "maintain consistency"). They are not so tied to specific systems, but require much interpretation to apply. In fact, their interpretation can - and often does - lead to prolonged debate. However, they can provide a useful framework for structuring an evaluation. Nielsen and Molich (1990) outline a small set of principles which provide the basis for training designers to use their Heuristic Evaluation method.

• An example of using guidelines in usability evaluation: a pre-release version of a visual interface to a UNIX operating system was evaluated against a set of 62 guidelines for usable software. They helped directly in identifying some usability problems, but more problems were identified from prior experience or as side-effects. The method was less effective than user-based evaluation. (See Jeffries et al 1991 for more details.)

Guidelines and Principles:
- can be used by designers and implementers throughout the design process
- are not user-based
- can help in identifying some usability problems
- increase designer awareness of usability issues
- can be ambiguous and difficult to interpret
- may be platform-specific

### 3.5.6. Heuristic Evaluation by Designers

Heuristic evaluation is a do-it-yourself 'expert' method, for use by members of a design team. It offers obvious advantages in terms of cost, administration and timing. At least as important is the fact that it removes the communication barrier between designer and evaluator. Its drawback is the difficulty in knowing with any degree of confidence how good or bad the resulting evaluation is.

Can designers be given sufficient basic expertise to perform cost-effective evaluations? Nielsen and Molich (1990) suggest that a short training session may be sufficient, equipping designers with a small number of usability principles that they can apply to analyse their design. They suggest that the method is intuitive, and that it is easy to motivate people to use it. It relies on pooling the results of evaluators. Heuristic evaluation by any individual designer after training may miss important flaws. Nielsen and Molich found that typically between 20% and 50% of usability problems were detected by individual heuristic evaluators. However, different evaluators detected different sets of problems. Hence Nielsen and Molich suggest that pooling the results of heuristic evaluation by several people from a design team can generate a comprehensive usability evaluation, at least in terms of the number of usability problems identified. One difficulty is the lack of any mechanism for evaluating which problems are important and which are trivial.

• Some examples of heuristic evaluation: Nielsen and Molich's method has been used in studies of interfaces to teledata and voice response systems, each study using a large number of 'designer/evaluators'.

Heuristic evaluation by designers:

- can be used relatively early in the design process
- is not user-based
- is cheap and quick
- employs simple, basic usability principles
- increases designer awareness of usability issues
- has broad scope, but fails to sift out unimportant problems
- relies on short training of designers for prediction of usability for end-users
- requires a large number of designer/evaluators to detect a spread of problems

## 3.6. OBSERVATIONAL EVALUATION

These are methods which involve *real people* using *working systems*. Observational methods of evaluation can range from almost entirely informal to highly structured: in essence, people are observed interacting with a system. Often their interaction is recorded in some way for later observation or analysis. To be helpful in evaluating the usability of the system, even the most informal approach to observation requires that the user has some work tasks to perform with the system. (Later, we shall consider co-operative evaluation methods, where the users contribute actively to the process of evaluation.)

Observational methods can offer extra benefits when designers and decision makers also take part in the observation. Designers of a system which has usability flaws may be more easily persuaded of the facts – that the system causes problems for users – when they see it with their own eyes. Managers, comparing possible alternative systems, or before accepting a system, get the opportunity to see just how usable each system is, and to talk issues through with professional evaluators and with the users.

### 3.6.1. Direct Observation

The simplest observational method takes the form of someone familiar with usability issues observing individual users performing tasks with the system. If the system is fully implemented, and in its natural work setting, then simple observation can tell us much about the use and usability of the system in that setting. Unfortunately – as most people are all too aware – knowledge of being observed can be intrusive, and can affect the way work tasks are performed. This shortcoming should not be seen as invalidating the method, though it should be borne in mind when interpreting such evaluations. Direct observation is also somewhat 'hit and miss': any detail which the observer may overlook is irretrievably lost. Observers have to analyse at the same time as they observe.

Direct observation is an important component of more sophisticated observational methods. It can be made more discreet in a specially constructed setting, such as a usability laboratory, where observers view through a two way mirror from an adjacent room, and may have the added advantage of close up views on monitor screens. Simple observation of users can give useful insights, and is especially valuable when used in conjunction with other evaluation methods, such as interview or empirical experiment.

Direct observation:
- requires an implemented system or prototype
- is user-based and involves user performance of work tasks
- is relatively simple to stage
- requires an expert observer to extract maximum relevant information
- can take place in a work setting
- may interfere with what is being observed

### 3.6.2. Video Recording

Making a video record of interaction between user and system offers real advantages. It also incurs some additional costs: in facilities required and, more significantly, in the time required to analyse the data, if the video record is to be analysed in detail. Some people may feel uneasy about 'being watched', and it is essential that recordings are made only with the informed consent of the users being filmed, but people generally are becoming less camera shy.

Most video recording and analysis of interaction takes place in usability laboratories, where evaluations are carefully staged. By using several cameras, it becomes possible to capture and synchronise data about screen display, hand movements and other user behaviour. Representative users perform specific work tasks with implemented systems, or with partially implemented prototypes which have sufficient functionality to allow realistic interaction. A major cost of such evaluation work has been the time required to analyse the data: in the past, up to 10 hours to analyse a single hour of recording, making the method relatively expensive. Several support tools are currently available or under development (see for example Rengger, 1991) which enable much more rapid analysis of video data.

With video, data capture can be separated from analysis. These two elements of evaluation can be performed in different locations and at different times, which is especially useful where the system being evaluated is designed for use in an industrial setting. The video is recorded on site, and the evaluator later identifies and analyses the relevant parts of the record.

The usability laboratory-based approach to evaluation has sometimes been criticised as being too detailed, and focusing on micro-issues while ignoring the wider work context. Clearly, evaluations must take account of the intended work context in which the system will be used. So long as this is done, then quite detailed analysis of user performance can offer distinct benefits, both for the certification of usability of software, and also during development. The evaluation of how people interact with prototypes allows us to identify and assess usability problems which would otherwise not become apparent before user testing of the fully implemented system.

It is good practice to involve designers, managers and future users in the evaluation, by giving them roles or simply inviting them into the viewing and control rooms of the usability laboratory. Their committed involvement can greatly assist in the shaping of the final system to meet the users' needs. Equally important, it can help the future end-users to feel that the system has been designed with their needs in mind. A list of leading UK usability centres and laboratories is given on page 105 of the *Usability Now!* HCI Tools and Methods Handbook.

• An example of use: a prototype control system for commissioning large industrial gas burners was evaluated with the participation of its designer and representative end-users (process control engineers). A simple simulation of control valves was sufficient for the users to role-play and perform selected on-site tasks, in the usability laboratory. Analysis of the video recordings revealed several usability problems which had not been apparent before the user-based evaluation, and contributed to specific design improvements.

Video recording:
• requires a working system or prototype
• is user-based and involves user performance of work tasks
• creates a reviewable record of interaction
• allows detailed analysis of interaction data
• can be time-consuming to analyse; faster when assisted by analytic support tools
• can be exploited to bring together managers, designers and end-users.

### 3.6.3. Interaction Monitoring

The idea of automated, effortless gathering of data about how people use systems is very appealing, and considerable research effort has been devoted to devising ways of automating the collection of objective, accurate data about interaction. There are some real difficulties, however, in producing relevant analyses of the captured data.

The principle of recording some kind of interaction log is well established. Many command line interfaces make provision for recording a trace of commands input and responses displayed.

• Typical uses: collections of logs – for example, of all use of a mainframe over an extended period – can be analysed statistically, to show the frequencies of use of different commands. An individual log can be analysed to reveal errors or inappropriate commands.

Graphical user interfaces present different problems for the collection and analysis of a log. The user is interacting directly with displayed objects: clicking on buttons, scroll bars, and menu bars, etc. The system is responding in subtle ways – for example, by changing the highlight of objects, and moving or displaying different parts of objects – as well as executing commands. If all the user and system actions occurring over a period of time are recorded, at a fine level of detail, then so much data may be collected that analysis becomes very difficult.

Two broad approaches can be taken to recording interaction with graphical user interfaces:

1. Replay based, with the aim of enabling sequences of interaction to be reviewed by analysts. An animated sequence of recorded screen display can give a visual recreation of events. In practice, this has not proved very satisfactory. It tends to swamp the analyst with raw data, while missing important aspects of the user's behaviour which would be captured by video-based evaluation.

2. Interaction records, which capture a simplified record of key user and system actions, in a form which can be interpreted by an evaluator. This approach can be particularly useful in the iterative development of a prototype.

• Typical uses: interaction records are widely used in development work and experimental studies with prototyping environments such as HyperCard (p 75 of the *Usability Now!* HCI Tools and Methods Handbook) and SuperCard. They require that the prototype is specifically programmed to record certain user and system actions, and to allow the evaluator access to this record. The key decision is which actions should be recorded. Too much data, especially to much low-level data, makes analysis difficult.

While monitoring can contribute quantitative data which would otherwise not be easily accessible, it is recommended that evaluation should generally have a broader base than a simple interaction record. The evaluator also needs information about the context and circumstances of the interaction.

Interaction monitoring
• requires an implemented system or substantially developed prototype
• is user-based and involves user performance of work tasks
• provides easy and inexpensive collection of data
• is usually unobtrusive
• can give a detailed time-stamped record of interaction
• can be facilitated by specific software tools for monitoring and analysis.
• generates data which can be difficult to analyse, because
  - there is often too much raw data
  - data may be too low level
  - there is a lack of information about why users acted as recorded

### 3.6.4. Co-operative Evaluation

Simple observation lacks access to the thoughts of the user, to the reasons why users did certain things, or failed to do them. The user's view can be invaluable in helping us to identify the aspects of a design which confuse or mislead, or fail to inform. Co-operative evaluation takes a major step beyond the simple observational methods described above, and perhaps deserves a section all its own. Co-operative methods do not rely solely on observing *real users* interacting with *working systems:* they *actively involve* those users in the process of evaluation.

One of the goals of involving users is to get them to provide insights into the reasons why they are doing what they are doing. But there is a problem with simply asking people to talk their way through a procedure while they carry it out, which co-operative evaluation methods must attempt to overcome: the very act of describing what they are doing can interfere with the way people work. They have to start thinking about the description, rather than concentrating on the doing. People also have many skills which they cannot completely describe in words. On balance though, these disadvantages are generally outweighed by the extra benefits which co-operative evaluation can deliver in terms of diagnostic information. For the method to be most effective, it is a good idea to give people guidance about the kind of information which is required, in order to get at the things most relevant to improving a design. It is quite legitimate for an evaluator to ask questions of the user during task performance – for example when the user is encountering a problem but not providing a commentary – so long as the questions are neutral and neither leading nor too distracting.

When it is considered important to avoid interfering with the way people work, an alternative is to ask people to describe retrospectively what they have done, and what the problems were. Here their difficulty may be in remembering accurately the things which were important at the time. Getting people to talk their way through a video recording of their own task performance can provide accurate reminders of what occurred. It gives us a comprehensive and potentially revealing record of task performance and the user's rationale, but can be costly in terms of the subjects' time and the analyst's effort.

A particularly elegant approach is described in Lucy Suchman's influential book, "Plans and Situated Actions" (Suchman, 1987). This involves analysis of the conversation between pairs of people trying to work out how to get a photocopier to do what they wanted. Lucy Suchman's work gives a surprisingly deep insight into the processes by which people explore their way towards finding how to use something.

Peter Wright and Andrew Monk have developed a method specifically for use by designers with little or no knowledge of Human Factors (see *Usability Now!* HCI Tools and Methods Handbook, page 99) in which users are actively encouraged to think aloud while working on set tasks with a prototype or a partial simulation. It is suggested that some kind of record should be made of each co-operative evaluation session, preferably on video, for later reviewing. There is a short training manual, the York Manual for Co-operative Evaluation (Wright, Monk and Carey, 1991), which many people have found a helpful guide. The method, as well as being validated in experiments, has been used with success in a US software development company for extensive evaluation work.

• An example of co-operative evaluation: the Wright and Monk method was used by a software development company to evaluate the interface of a relational document retrieval system, before a planned extension of its functionality. The aim was to make the system more powerful without destroying its popularity or ease of use. Co-operative evaluation was found to give many benefits for minimal risk, and as a result was incorporated into the standard company procedures.

Co-operative evaluation:
• requires an implemented system, partially implemented prototype or mock-up
• is user-based
• gives good access to relevant views of users
• is aimed at identifying problems which users encounter
• provides diagnostic, not quantitative evaluation
• can bring together designers and end-users

## 3.7. SURVEY EVALUATION

Survey methods give the evaluator access to the users' subjective views of a system. They involve *real users*, but information is not gathered while the users are interacting with the system.

### 3.7.1. Questionnaires

Questionnaires and Surveys are widely used in evaluation. They can give very cost-effective access to data about the usability of a system. They offer the real advantage of providing data about the end-user's views of system quality, rather than some expert or theoretician's view, and can provide data about system usability in a real work setting. However, the data gathered may be of variable quality. It is essential to ensure that the group of people responding to the questionnaire or survey accurately match the actual or intended group of end-users, and that they have used the software for work purposes, performing the specific work tasks in which you are interested.

Closed questions can constrain answers to yes / no, or to a point on a scale of preference. (See p 115 of the *Usability Now!* Guide to Usability for a useful discussion of questionnaire design). Questionnaires can provide data which are readily analysed, but require careful construction to avoid ambiguity and to gather data of the required scope about the system and its use. Closed questions do not allow the respondent to express views about any other aspect of the system's usability, even if it seems important to the user. Open questions such as 'What would you most like to change about this system to enhance your use of it?' would allow the expression of relevant criticism, but can be difficult to analyse. It is widely accepted that clearly focused questions about actual system experience are the most likely to yield useful information. It is best to use a commercially available questionnaire, with proven validity and reliability, if you can find one which meets your specific needs.

The kinds of factors which questionnaires can inform us about are illustrated by two examples. SUMI, the Software Usability Measurement Inventory (Porteous and Kirakowski; 1991) has been developed for quantitative measurement of how usable a product is in the eyes of the user. SUMI gives information about five factors: Efficiency, Affect, Helpfulness, Control and Learnability, plus a Global measure of usability. Its validity and reliability have been established in field trials, and SUMI offers the prospect of inexpensive collection of trustworthy data about the usability of a product. There is a handbook to support the correct use of the questionnaire. SUMI is the successor to the questionnaire described in the *Usability Now!* HCI Tools and Methods Handbook, page 111. Another questionnaire for evaluating user interface satisfaction is QUIS (Chin, Diehl, and Norman; 1988. See *Usability Now!* HCI Tools and Methods Handbook, page 106). QUIS has achieved high reliability in evaluating systems in terms of four factors: Learning; Terminology and Information Flow; System Output and System Characteristics.

• An example of the use of a questionnaire: SUMI was used by a telecommunications company to evaluate data entry systems. The company was changing to a UNIX based system with a network of SUN workstations, with a total of about forty users. They ran a comparative trial for two months of two different data entry systems, and used SUMI to evaluate their usability. One system came out significantly better than the other. Happily, this was also the system preferred by the DP manager!

Questionnaires:
- require an implemented system or prototype
- are user-based
- can be highly cost-effective
- can yield subjective views about use of system in work context
- constrain the nature of data gathered (by asking some questions, but not others)
- can only gather retrospective views of usability

### 3.7.2. Interviews

Interviews are much more time-consuming than questionnaires, if a large sample of users is to be included in the survey. They require careful pre-planning, and a good degree of expertise in the interviewer, to ensure that the questions cover all the relevant topics, and are not delivered in a way which unduly influences the nature of the reply.

A structured interview, with a pre-determined set of precisely phrased questions, is required if data are to be analysed statistically.

A flexible interview, covering pre-specified topics but in a style and order shaped by the responses of the user, can elicit more revealing information about the usability of a system. It allows interviewers to pursue interesting lines of enquiry (assuming their HCI expertise and powers of detection are sufficiently developed). Results may be more difficult to analyse en masse, but individually can be highly informative. Some people choose to record interviews. Analysing large numbers of audio or video recordings of interviews is generally too time consuming to be cost-effective, but audio recordings can be invaluable for retrieving illuminating quotes verbatim. The power of one or two good quotes in convincing decision-makers about the qualities (or defects) of a system should never be underestimated.

Interviews:
- require an implemented system or prototype
- are user-based
- can elicit helpful information about users' views
- if structured
  - require careful planning and design to cover all issues
  - can yield statistically analysable results
- if flexible
  - require interviewer with both interviewing skills and HCI knowledge
  - are difficult to analyse
  - can provide revealing insights

## 3.8. EXPERIMENTAL EVALUATION

What would be the effect on usability of changing a given factor in a design? This is the kind of question which controlled experimental studies can answer. For people with a scientific training, soundly designed empirical experiments have immediate appeal for the testing of hypotheses about the effects of various factors in a design. They can be a source of sound, objective and valid data, so long as all other variable factors can be controlled.

Simple controlled experimental studies, for example of simple physical systems, can be relatively straightforward to design and conduct. With such systems, it is feasible to control everything which might influence the outcome of the experiment. Unfortunately, things are more complicated when people are part of the experiment. The diversity of factors influencing their behaviour, including social surroundings, makes it extremely difficult for an experimenter to know how well the experimental conditions have been controlled. The problem is of such complexity that many investigators spend their professional lives grappling with questions of how to design and analyse such experiments. Monk (1985) provides a thoughtful assessment of the value of controlled experiments, and an introduction to key issues in designing experiments specifically for evaluation. A substantial part of many undergraduate courses in the behavioural sciences is concerned with equipping students to design, conduct and analyse valid controlled experimental studies. Badly designed studies are misleading. Properly designed controlled experimental studies can contribute to the advance of scientific knowledge, but can be very time-consuming and expensive, and usually leave the experimenter not knowing how much the results can be generalised to other contexts.

This means that controlled experimental studies may fail to meet some of the principal requirements for practical usability evaluations during the development of a system. Where they can be useful is in testing the usability of alternative designs of discrete components of a system, capable of being tested separately from the system as a whole, such as input devices and icon sets. Even then there is some difficulty in knowing how well, for example, an experimental subject's understanding of an icon will match the understanding a user will have of that icon when it is embedded in the context of using a specific system for real work tasks in a specific workplace.

Controlled experimental studies for usability evaluation:
• are user-based, and can compare usability for different groups of user
• can provide quantitative data about specific factors in a design
• can produce results with good validity and reliability
• require careful design by a competent theoretician
• are usually expensive, time-consuming, and difficult to fit into system development
• give results which may be difficult to generalise, because of limitations necessarily imposed by the experimental design

# 4. INFORMATION SOURCES

## 4.1. FURTHER READING

### 4.1.1. Helpful Overviews

Apple Computer (1987). Human Interface Guidelines: The Apple Desktop Interface. Wokingham, England, Addison-Wesley; ISBN 0 201 17753 6.

Booth,P.A. (1989). An Introduction to Human-Computer Interaction. Hillsdale, NJ, USA, Lawrence Erlbaum Associates; ISBN 0 86377 1238.

Karat,J. (1988). Software Evaluation Methodologies. In: Helander, M. (Ed.) Handbook of Human-Computer Interaction. Elsevier (North-Holland), pp 891-903; ISBN 0 444 88673 7.

Monk,A. (1985). Fundamentals of Human-Computer Interaction; London, Academic Press; ISBN 0 12 504 582 4.

Kirakowski,J. and Corbett,M. (1990). Effective Methodology for the Study of HCI. Amsterdam, Elsevier (North-Holland); ISBN 0 444 88447 5.

Lea,M. (1988). Evaluating User Interface Designs. In: Rubin,T., User Interface Design for Computer Systems. Chichester, UK, Ellis Horwood; ISBN 0407 21172 5.

Shackel,B. and Richardson,S. (Eds) (1991). Human Factors for Informatics Usability; Cambridge, UK, Cambridge University Press; ISBN 0 521 36570 8.

Snyder,K.M. (1991). A Guide to Software Usability. New York, IBM Corporation, Ref SC26-3000-00.

Whiteside,J., Bennett,J. and Holtzblatt,K. (1987). Usability Engineering: Our Experience and Evolution. In Helander, M. (Ed.) Handbook of Human-Computer Interaction. Amsterdam, Elsevier (North-Holland) pp 791-817; ISBN 0 444 88673 7.

### 4.1.2. Guides to Specific Methods

Clegg,C., Warr,P, Green,T., Monk,A., Kemp,N., Gordon,A. and Lansdale,M. (1988). People and Computers: How to Evaluate Your Company's New Technology. Chichester, UK, Ellis Horwood; ISBN 0 7458 0419 5.

Fitter,M., Robinson,S., Rector,A., Horan,B., Nowlan,A., Wilson,A. and Newton,P. (1991). Supportive Evaluation Handbook: A Methodology to Facilitate System Development in the Healthcare Domain. MRC/ESRC Social and Applied Psychology Unit, University of Sheffield.

Polson,P.G., Lewis,C.H, Rieman,J. & Wharton,C. (1991). Cognitive Walkthroughs: A Method for Theory-Based Evaluation of User Interfaces. Tech Report 91-1, Institute of Cognitive Science, University of Colorado, Boulder, Colorado.

Porteous,M.A. and Kirakowski,J.Z. (1991). Software Usability Measurement Inventory (SUMI) User Handbook. Human Factors Research Group, University College Cork, Ireland.
[Fax: +353 21 270439 Telephone: +353 21 276871 x2636]

Ravden,S.J. and Johnson,G.I. (1989). Evaluating Usability of Human-Computer Interfaces: A Practical Method. Chichester, UK, Ellis Horwood, ISBN 0 7458 06147.

Wright,P.J., Monk,A.F. & Carey,T. (1991). Co-operative Evaluation: the York Manual. Department of Psychology, University of York, YORK, YO1 5DD, UK. [Telephone: 0904 433148]

### 4.1.3. Reports of Evaluation Applied in Industry

Crellin,J., Horn,T. and Preece,J. (1990). Evaluating Evaluation: A Case Study of the Use of Novel & Conventional Evaluation Techniques in a Small Company. In Diaper,D.,Gilmore,D., Cockton,G. & Shackel,B., Human-Computer Interaction, Proceedings of INTERACT'90, 3rd IFIP International Conference on HCI. North-Holland, pp 329-336.

Jeffries,R., Miller,J.R., Wharton,C. and Uyeda,K.M. (1991). User Interface Evaluation in the Real World: A Comparison of Four Techniques. In Robertson, S.P., Olson, G.M., and Olson, J.S., CHI '91 Conference Proceedings: Reaching through Technology. ACM, New York, pp 119-124.

Mulligan,R.M., Altom,M.W. and Simkin,D.K. (1991). User Interface Design in the Trenches: Some Tips on Shooting from the Hip. In Robertson, S.P., Olson, G.M., and Olson, J.S., CHI '91 Conference Proceedings: Reaching through Technology. ACM, New York, pp 232-236.

### 4.1.4. Academic Views of Evaluation

Bellotti,V. (1990). A Framework for Assessing Applicability of HCI Techniques. In Diaper,D.,Gilmore,D., Cockton,G. & Shackel,B., Human-Computer Interaction, Proceedings of INTERACT'90, 3rd IFIP International Conf. on HCI. North-Holland. pp 213-218.

Denley,I. and Long,J.B.(1990), A Framework for Evaluation Practice. In: E.J. Lovesey (ed), Contemporary Ergonomics 1990. London; Taylor and Francis.

Howard,S. and Murray,D.M. (1987). A Taxonomy of Evaluation Techniques for HCI. In: Bullinger,H.-J. and Shackel,B. (Eds) Human-Computer Interaction - INTERACT'87, (Proceedings of 2nd IFIP Conference on HCI). Elsevier, Amsterdam. pp 453-459.

Vainio-Larsson,A. and Orring,R. (1990). Evaluating the Usability of User Interfaces: Research in Practice. In Diaper,D., Gilmore,D., Cockton,G. & Shackel,B., Human-Computer Interaction, Proceedings of INTERACT'90, 3rd IFIP International Conf. on HCI. North-Holland. pp 323-328.

Whitefield,A., Wilson,F., and Dowell,J. (1991). A Framework for Human Factors Evaluation, Behaviour and Information Technology, 10(1), 65-80.

### 4.1.5. *Usability Now!* Publications

The Design Council (1991). Directory of HCI Consultants.

The HCI Service (1991). HCI Tools and Methods Handbook.

J&G Consultants (1991). Directory of HCI Training.

Preece,J., Davis,G. and Keller,L. (Eds.) (1990). A Guide to Usability, The Open University in association with the Department of Trade and Industry.

Stewart, T. (1991). Directory of HCI Standards.

• *Usability Now!* Enquiry Point: 0509-264083

## 4.2. REFERENCES


Apple Computer (1987). Human Interface Guidelines: The Apple Desktop Interface, Wokingham, England, Addison-Wesley; ISBN 0 201 17753 6.

Bösser,T., and Melchior,E.M. (1990). Cognitive Modelling, Rapid Prototyping, and User-Centred Design with the SANE Toolkit. In Commission of The European Community (Ed.), ESPRIT '90 Conf Proc., Kluwer, pp 589-606. [Bösser contact details: Fax: +49 251 37223 Telephone +49 251 834125]

Card,S., Moran,T.P., and Newell,A. (1980). The Keystroke-Level Model for User Performance Time with Interactive Systems. Communications of the ACM 23,7 (July) pp 396-410.

Chin,J.P., Diehl,V.A. and Norman,K.L. (1988). Development of an Instrument Measuring User Satisfaction of the Human-Computer Interface. In CHI '88 Conference Proceedings, ACM, New York, pp 213-218.

Kieras,D.E. (1988). Towards a Practical GOMS Model Methodology for User Interface Design, in Helander,M., (Ed.), Handbook of Human-Computer Interaction, pp 135-157.

Lewis,C., Polson,P., Wharton,C. & Rieman,J. (1990). Testing a walkthrough methodology for theory-based design of walk-up-and-use interfaces. In Chew, J.C. & Whiteside, J. (Eds.), CHI'90 Conference Proceedings: Empowering People, ACM, pp 235-241.

Monk,A. (1985). How and When to Collect Behavioural Data. In: Monk, A., Fundamentals of Human-Computer Interaction; London, Academic Press, pp 69-79.

Nielsen,J. and Molich,R. (1990). Heuristic Evaluation of User Interfaces. In Chew,J.C. & Whiteside,J. (Eds.), CHI'90 Conference Proceedings: Empowering People, ACM, pp 249-256.

Porteous,M.A. and Kirakowski,J.Z. (1991). Software Usability Measurement Inventory (SUMI) User Handbook, Human Factors Research Group, University College Cork, Ireland. [Fax: +353 21 270439 Telephone: +353 21 276871 x2636]

Preece,J., Davis,G. and Keller,L. (Eds.) (1990). A Guide to Usability, The Open University in association with the Department of Trade and Industry.

Rengger,R. (1991). Indicators of usability based on performance; In: Bullinger HJ (Ed.) Proceedings of the 4th International Conference on Human Computer Interaction, Stuttgart, September 1991. Elsevier, pp 656-660.

Ravden, S.J. and Johnson, G.I. (1989) Evaluating Usability of Human-Computer Interfaces: A Practical Method, Ellis Horwood, ISBN 0 7458 06147.

Rieman,J. Davies,S., Hair,C., Esemplare,M., Polson,P. & Lewis,C. (1991). An Automated Cognitive Walkthrough. In Robertson, S.P., Olsen, G.M. & Olsen, J.S. (Eds.), Human Factors in Computing Systems: Reaching through Technology, CHI'91 Conf Proceedings, ACM, pp 427-8.

Suchman,L.A. (1987). Plans and Situated Actions: The Problem of Human-Machine Communication, Cambridge, Cambridge University Press.

Stewart, T. (1991). Directory of HCI Standards. A DTI *Usability Now!* Publication.

Wright,P.J., Monk,A.F. & Carey,T. (1991). Co-operative Evaluation: the York Manual, Department of Psychology, University of York, YORK, YO1 5DD, UK. [Telephone 0904 433148]