Behaviour and Information Technology, **13**, 132-145 (1994)

Usability measurement in context

Nigel Bevan and Miles Macleod

National Physical Laboratory, Teddington, Middlesex, UK

email: Nigel@hci.npl.co.uk and Miles@hci.npl.co.uk

Keywords: usability, measurement, evaluation, metrics, standards, guidelines

Abstract

Different approaches to the measurement of usability are reviewed and related to definitions of usability in international standards. It is concluded that reliable measures of overall usability can only be obtained by assessing the effectiveness, efficiency and satisfaction with which representative users carry out representative tasks in representative environments. This requires a detailed understanding of the context of use of a product. The ESPRIT MUSiC project has developed tools which can be used to measure usability in the laboratory and the field. An overview is given of the methods and tools for measuring user performance, cognitive workload and user perceived quality.

Contents

© Crown Copyright 1993

1. Benefits of improved usability

Most computer software in use today is unnecessarily difficult to understand, hard to learn, and complicated to use. Difficult to use software wastes the user's time, causes worry and frustration, and discourages further use of the software. Why is the usability of most computer software so poor, and what are the benefits that more usable software could bring to the employer and supplier?

Benefits to the employer include:

- Usable software increases productivity and reduces costs. Difficult to use software is time consuming to use, and not exploited to full advantage as the user may be discouraged from using advanced features. Difficult to learn software also increases the cost of training and of subsequent support.

- Usable software increases employee satisfaction. Difficult to use software reduces motivation and may increase staff turnover.

- In Europe, employers have an obligation to meet the requirements of the Display Screen Equipment Directive (CEC 1990) which requires software in new workstations to be "easy to use" and to embody "the principles of software ergonomics" (see Bevan, 1991a).

Benefits to the supplier include:

- Software suppliers are increasingly facing a market where users demand easier to use software, and legislation and standards are putting pressure on employers to provide usable software. End-users are becoming more discerning. Promotional programmes, such as "Usability Now!" in the UK, have made purchasers more conscious of the benefits of usability, and more inclined to give greater weight to ease of use when making purchases.

- Usability is increasingly providing suppliers with a market edge, as can be seen in recent advertising campaigns by Microsoft and Amstrad which have promoted ease of use as a major selling feature.

- International standards for usability and the user interface are nearing finalisation and will increasingly be cited in public procurement, and provide a means to meet the requirements of the European Display Screen Equipment Directive (Bevan 1991b).

1.1. Usability evaluation and system development

Given the potential benefits to the supplier, employer and end user, why is so much software so difficult to use? A major problem is that in spite of recent acknowledgement that usability is an important software quality (e.g. ISO 9126), it has remained a fuzzy concept which has been difficult to evaluate and impossible to measure. As a consequence it is often not explicitly identified as part of the user requirements and does not form part of the product specification. Even when ease of use has been identified as a desirable property, how can a product developer with the responsibility for developing a product to specification, on time and within budget justify spending the extra resources required to produce a usable product? An additional problem is that dealing with usability requires skills in human factors, and is difficult to integrate with many existing design processes.

What is required is a common understanding of what constitutes usability and how it can be specified and measured. This has been one of the major objectives of the development of ISO 9241-11: Guidance on specifying and measuring usability (Brooke et al, 1990), which will be published as a Draft International Standard in 1994.

In parallel with this activity, the ESPRIT MUSiC project has been developing a set of methods for specifying and measuring usability, which are consistent with the approach taken in the standard (see section 6 below). The objectives of measuring usability during design are: to ensure that the delivered product reaches the minimum required level of usability, to provide feedback during design on the extent to which the objectives are being met, and to identify potential usability defects in the product.

2. Usability features and attributes

2.1. Can usability be measured in terms of features and attributes?

The ideal way to specify and measure usability would be to specify the features and attributes required to make a product usable, and measure whether they are present in the implemented product. This is the approach taken with other software qualities such as functionality, efficiency and portability, and it enables quality to be designed into a product.

The problem with usability is that it is very difficult to specify what these features and attributes should be, in particular because the nature of the features and attributes required depends on the context in which the product is used. There have been many attempts to describe these features and attributes, including dialogue principles, guidelines and checklists, and analytic procedures.

2.2. Dialogue principles

High level principles for user interface design are contained in ISO 9241-10 (ISO 1993b) "Dialogue principles" (which is based on an earlier DIN standard). These principles are: suitability for the task, suitability for learning, suitability for individualisation, conformity with user expectations, self descriptiveness, controllability, and error tolerance.

ISO 9241-10 gives applications and examples of each principle. For example, one application of controllability is "If interactions are reversible and the task permits, it should be possible to undo the last dialogue step". An example of this is "The dialogue system offers the possibility to access deleted objects".

These principles have broad application, and are particularly relevant in interpreting the requirement in the European Display Screen Equipment Directive (CEC, 1990) that "the principles of software ergonomics shall be applied" to the operator/computer interface. However it has proved impossible to formally assess compliance to these type of general principles. In particular it is difficult to decide whether and to what extent they apply in borderline cases. This means that they cannot be used as a basis for measurement.

2.3. Guidelines and checklists

There are many user interface guidelines which can improve usability when applied in design. Some guidelines are in terms of user interface features (e.g. provision of help, screen layout of a menu), and others state higher level attributes (e.g. consistency, flexibility). There are a number of collections of guidelines for the design of user interfaces. The best known is by Smith and Mosier (1986), and this formed the starting point for more rigorous guidelines which are being published by ISO as international standards (consisting primarily of recommendations) (ISO 1993a). The most ambitious example of a complete specification of features is probably the Menu Dialogue guidelines in ISO 9241-14 (ISO 1993d). These are in the form of 112 conditional recommendations. Only those guidelines which are applicable have to be followed, for instance: "If the ordering of option usage is known, options should be placed in that order in the menu."

Other more general checklists include those by Ravden and Johnson (1989) where users fill in detailed checklists about the acceptability of various aspects of the interface thus highlighting particular types of problems; and EVADIS (Oppermann et al 1989, Reiterer 1992) where usability specialists evaluate the usability of the system for pre-defined tasks by assessing whether it meets detailed requirements given in checklists.

Some procedures for the assessment of the usability of software (e.g. McGinley and Hunter, 1992) also use adherence to guidelines as a basis for assessment.

There are also more detailed style guides. For example, when a graphical interface is appropriate, the IBM CUA Guide to User Interface Design (1991a, 1991b) and the Windows Interface Design Guide (1992) provide detailed advice on how the features can be most effectively implemented.

2.4. Limitations of guidelines and checklists

Guidelines have the advantage that they can be applied early in design, and adherence to most guidelines can be assessed merely by inspection of the product without user testing. However, guidelines have a number of other limitations:

- Detailed and specific guidelines are likely to be appropriate only for specific systems and specific types of users. However, guidelines and principles expressed in general terms are difficult for developers and evaluators to interpret – they can mean very different things to different people. (This is eloquently demonstrated by Grudin, 1989).

- There is no guarantee that any particular set of guidelines is exhaustive and deal with all relevant aspects of the user interface.

- Many alternative design solutions can be equally compatible with guidelines, but changing an interface feature to be compatible with one guideline often makes it incompatible with another. There is no easy way to trade off the benefits of different guidelines.

- The effectiveness with which guidelines are applied depends on the skill of the designer in interpreting and applying them and making any necessary trade-offs.

- Some usability attributes are context-dependent properties of interaction which can only be evaluated when a product is actually used by representative users for representative tasks.

- Evaluating whether a product is consistent with guidelines can be very time consuming. For instance, applying the menu guidelines standard to a complete product requires that every menu in a product is checked for conformance with every applicable recommendation.

- Following guidelines does not ensure that a product reaches any particular level of usability. In particular the structure of dialogue design is just as important as the more easily assessable surface features. In many cases the usability of a product will be improved by redesigning the interface to be consistent with guidelines, but a much bigger improvement to the usability can often be made by considering whether the task can be carried out more effectively by a more fundamental redesign (e.g. avoiding the use of menus to search for information by supplying a unique key which gives direct access).

- Guidelines often attempt to generalise across a wide range of characteristics of users, tasks and environments, and it is very difficult to specify rigorously the limits of the context in which a guideline is applicable.

- Where guidelines are expressed in general terms, their interpretation may rely so much on expert opinion that objective evaluation is simply not possible.

- When guidelines are used for evaluation the result is a checklist showing which parts of a product conform to which recommendations. While this can be used to identify potential problems with the interface, it cannot be turned into measures as there is no way to accurately weight the importance of the different recommendations.

3. Analytical evaluation

More formal analytical approaches to evaluation can produce measures which give predictions about usability before implementation has started. The methods require at least a partial specification of the system and its user interface, and employ some kind of theoretical representations of the user. They can be quite simple and of limited scope – for example the Keystroke Level Model (K-LM: Card, Moran and Newell, 1980) – or more complex and wide-ranging, such as SANe (Gunsthövel and Bösser, 1991). Being based on specifications, they can be conducted early in the design cycle. Analytic methods only model limited aspects of users, tasks and software, and are concerned primarily with performance rather than issues of user satisfaction. The results tend to focus on quite detailed aspects of the design, such as menu design and specific dialogue sequences. While they cannot assure usability, they can be very effective at identifying significant problems early in design.

A Keystroke-Level Model analysis describes interaction at the level of individual keystrokes and mouse movements. 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. The accuracy of the K-LM is limited by the fact that it does not take account of contextual and higher cognitive issues. However, the rapid growth of GUI development has revealed that low level analysis is often useful. For example, some applications are being developed which force users to perform cumbersome sequences of mouse actions, to carry out simple operations. A K-LM analysis of such operations can identify the potential advantage in actions and time saved of providing single keystroke alternatives for frequently performed operations.

More sophisticated analytic models can be employed, early in design, to test specifications for specific aspects of usability for users who have different degrees of skill and knowledge. These models represent at some level the processes of the system and the cognitive processes or abilities of the user. They work by generating predictions of the complexity of thought and action which are demanded of the user in performing specified tasks. They require a cognitive model of the user, and a specification of the user system interface. Their aim is to reduce the need for design changes later in development, when alterations are more difficult and expensive to make. Currently available methods require substantial expertise to deliver useful predictions, although some recent methods are easier to apply, such as the more recent version of Cognitive Complexity Theory (Kieras 1988) and SANe (Gunsthövel and Bösser, 1991).

The SANE approach uses a dynamic model of the user interface, and separate models of user tasks. User procedures (the way in which the user solves the tasks with a specified system) can be generated by simulation. The measures produced describe the efficiency of use, learning requirements, cognitive workload, adaptedness of the system to the tasks, and robustness. The separation of device and task models facilitates comparison of alternative design options, and the usability of the device for different tasks. The SANe toolkit, developed partly within the MUSiC Project, is a development environment for building the models, simulating the procedures, and deriving measures.

4. Usability and context of use

It is not meaningful to talk simply about the usability of a product, as usability is a function of the context in which the product is used. The characteristics of the context (the users, tasks and environment) may be as important in determining usability as the characteristics of the product itself. Changing any relevant aspect of the context of use may change the usability of the product. For instance, the user interface may be improved by conforming to good dialogue design practices, or the fit between the user and the rest of the overall system may be improved through means such as selection and training of users or good task design. A product which is usable by trained users may be unusable by untrained users. Aspects of the working environment such as lighting, noise, or workstation design may also affect usability.

4.1. Relationship of the dialogue principles to the context of use

The dialogue principles in ISO 9241-10 cannot be used for design or evaluation without first identifying the context of use. Some of the dialogue principles are closely related to specific aspects of the context of use. "Suitability for the task" deals with design issues which are closely associated with the task characteristics. When applying this principle particular consideration should be given to those tasks which particular types of user may need to perform to meet the goals of the user organisation. "Suitability for learning", "suitability for individualisation", and "conformity with user expectations" deal with design issues which are closely associated with the user characteristics. When applying these principles particular consideration should be given to the needs of different types of intended users when performing intended tasks in particular situations.

4.2. Usability is the quality of use in a context

Usability is a property of the overall system: it is the quality of use in a context. As explained above, existing methods for predicting usability are limited in their accuracy as they only model limited aspects of the users, the tasks and environments. Quality of use can instead be measured as the outcome of interaction in a context: the extent to which the intended goals of use of the overall system are achieved (effectiveness); the resources such as time, money or mental effort that have to be expended to achieve the intended goals (efficiency); and the extent to which the user finds the overall system acceptable (satisfaction). The overall system consists of the users, tasks, equipment (hardware, software and materials) and physical and organisational environments which influence the interaction.

Usability in this sense is defined in ISO 9241-11 (ISO 1993c) as the quality of use:

> "The effectiveness, efficiency and satisfaction with which specified users achieve specified goals in particular environments".

The remainder of this paper explains and extends the approach to usability taken in ISO 9241-11 (Brooke et al 1990). The relationship between the factors is illustrated in Figure 1.

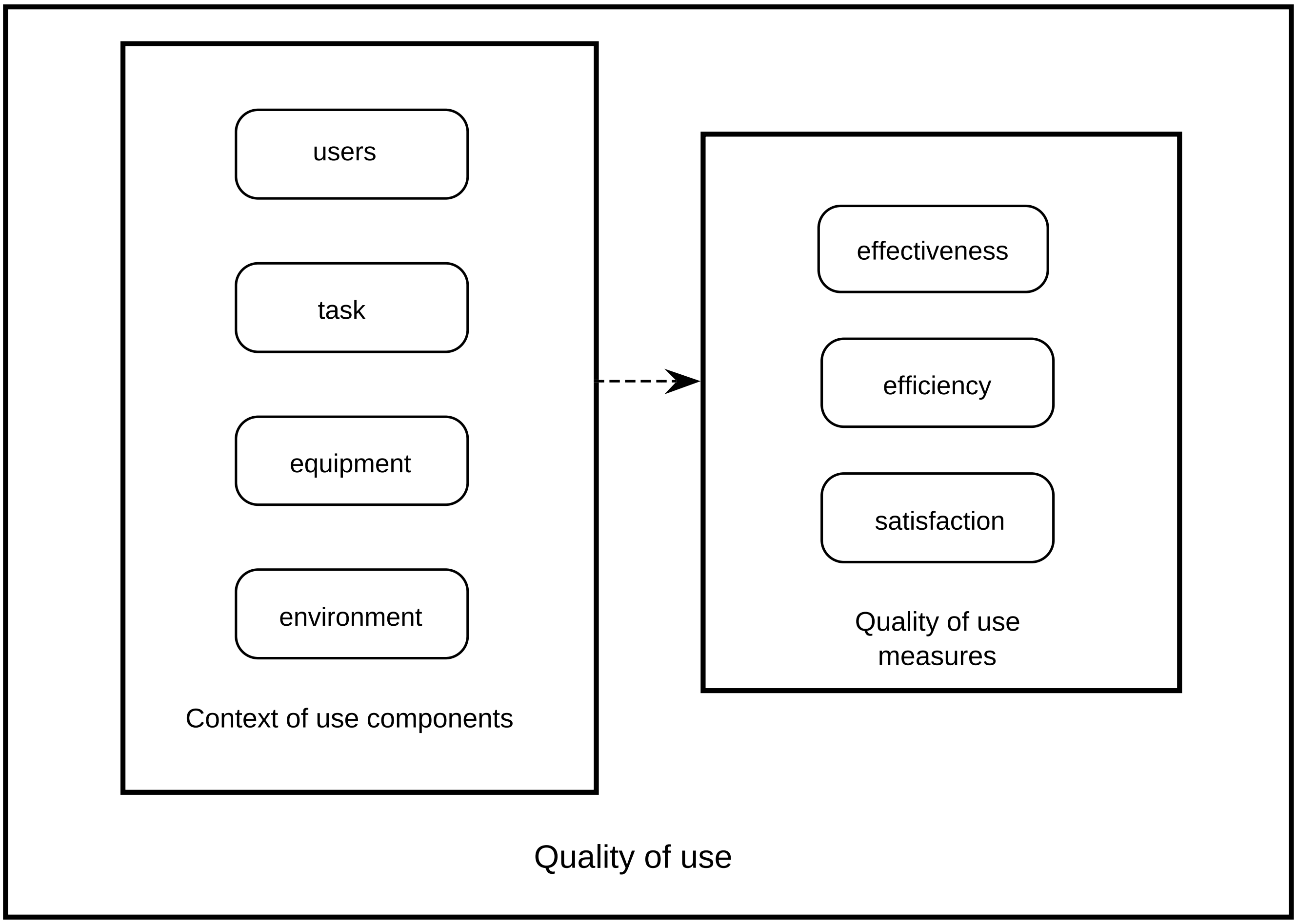

Figure 1 - Usability Factors

4.3. Usability attributes of a product and usability of an overall system

The quality of use of an overall system encompasses all factors which may influence use of a product in the real world, including organisational factors such as working practices and the location or appearance of a product, and individual differences between users such as those due to cultural factors and prejudice. This broad approach has the advantage that it concentrates on the real purpose of design of a product - that it is usable by real users carrying out real tasks in a real technical, physical and organisational environment.

The term usability is sometimes used more narrowly to refer specifically to the usability attributes of a product, e.g. the definition of usability as a software quality in ISO/IEC 9126 (ISO 1992):

> "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".

The relationship between the definitions is that the product-centred definition is concerned with the usability attributes of the software which will determine usability in a specific context. The usability attributes in the product contribute to the effectiveness, efficiency and satisfaction with which specified users achieve specified goals in particular environments. But the effectiveness, efficiency and satisfaction will also depend on other software qualities such as functionality, reliability and system efficiency, in addition to the relevant aspects of the context of use. The usability attributes of a product are thus only one contribution to the quality of use of an overall system. The usability of a product can thus be defined as:

> "The ability of a product to be used with effectiveness, efficiency and satisfaction by specified users to achieve specified goals in particular environments."

In practice the term usability means different things to different people. Usability can be viewed in different ways for different purposes, focusing on one or more of the following complementary views:

a) the product-centred view of usability: that the usability of a product is the attributes of the product which contribute towards the quality of use;

b) the context of use view of usability: that usability depends on the nature of the user, product, task and environment;

c) the quality of use view of usability: that usability is the outcome of interaction and can be measured by the effectiveness, efficiency and satisfaction with which specified users achieve specified goals in particular environments.

The designer and specifier of a product need to start with the context of use view to identify the context in which the product will be used. All the components of the overall system that influence the quality of use must be described if it is to be fully specified or evaluated. Given this context, the developer will be concerned with maximising usability and diagnosing defects by taking the product view. Both specifier and developer will wish to improve the quality of use. Since this cannot be directly assessed they can use measures of effectiveness, efficiency and satisfaction to specify usability and measure whether desired levels are achieved.

There are other approaches to usability evaluation which concentrate on identifying usability defects. Defects can be identified by expert assessment, aided by guidelines and checklists (e.g. heuristic evaluation) or by user-based testing (Nielsen, 1993) . When these two approaches are optimised they can be equally cost effective in improving design (Nielsen and Mack, 1994), although user-based testing is generally better at identifying more serious defects (Karat et al 1992). This paper emphasises the benefits of using user-based testing to measure usability, so that in addition to identifying usability defects, usability can be specified and evaluated.

5. Specifying and measuring usability

5.1. Context of measurement must match context of use

When usability is measured, it is important that the conditions for a test of usability are representative of important aspects of the overall context of use. Unless evaluation of usability can take place in conditions of actual use, it will be necessary to decide which attributes of the actual or intended context of use are to be represented in the context used for evaluation. When specifying or evaluating usability it is therefore important that the context selected is representative of the important aspects of the actual or intended context of use. Particular attention should be given to those attributes which are judged to have a significant impact on the quality of use of the overall system.

Using a breakdown of the context such as the example given in Table 1 (based on Maissel et al, 1991), information needs to be collected under each of the headings on the context in which the equipment is actually used (or is intended to be used).

USERS	TASK	ENVIRONMENT
Personal details User types Audience and secondary users **Skills & knowledge** Product experience System knowledge Task experience Organisational experience Training Keyboard & input skills Qualifications Linguistic ability General knowledge **Personal attributes** Age Gender Physical capabilities Physical limitations and disabilities Intellectual ability Attitude Motivation	Task breakdown Task name Task goal Task frequency Task duration Frequency of events Task flexibility Physical and mental demands Task dependencies Task output Risk resulting from error	**Organisational Environment** **Structure** Hours of work Group working Job function Work practices Assistance Interruptions Management structure Communications structure Remuneration **Attitudes & culture** Policy on use of computers Organisational aims Industrial relations **Job design** Job flexibility Performance monitoring Performance feedback Pacing Autonomy Discretion
	EQUIPMENT	**Technical environmen**

	Basic description Product identification Product description Main application areas Major functions **Specification** Hardware Software Materials Other Items	**Configuration** Hardware Software Reference materials **Physical environment** **Workplace conditions** Atmospheric conditions Auditory environment Thermal environment Visual environment Environmental instability **Workplace design** Space and furniture User posture Location **Workplace safety** Health hazards Protective clothing & equipment

Table 1 Example of Breakdown of Context

5.2. Choosing tasks, users, environments

The choice of appropriate types of user, tasks and environments depends on the objectives of the evaluation, and how the product is expected to be used. For a general-purpose product it may be necessary to specify or measure usability in several different contexts, which will usually be a subset of the possible contexts and of the tasks which can be performed.

Care should be taken in generalising the results of any measurement of usability to another context which may have significantly different types of users, tasks or environments. Also, if usability is measured over a short period of time it may not take account of infrequent occurrences which could have an impact on usability, such as intermittent system errors. For a general-purpose product it will usually be necessary to specify or measure the quality of use in several different representative contexts, which will be a subset of the possible contexts and of the tasks which can be performed. There may be differences between the quality of use in these contexts.

The nature of the product and tasks being specified or evaluated will determine the range of the measures and the breadth of the context which may influence the measures. For instance the usability of a new cut and paste feature will be influenced by a comparatively limited and well-defined context, while a new time management system will be influenced by a wide context which may include other users and organisational issues. For the cut and paste feature efficiency might be based on the mental effort or time required to successfully complete tasks, while for the time management system, efficiency might be based on time or overall financial cost.

5.3. Choosing usability measures

A description of the quality of use should consist of appropriate measures of user performance (effectiveness and efficiency), and of user satisfaction. Because the relative importance of components of usability depends on the context of use and the purposes for which usability is being described, there is no general rule for how measures can be combined. It is normally necessary to provide at least one measure for each of the components of quality of use, and it will often be necessary to repeat measures in several different contexts.

5.3.1. Effectiveness

Measures of effectiveness relate the goals or sub-goals of using the system to the accuracy and completeness with which these goals can be achieved.

For example if the desired goal is to transcribe a 2-page document into a specified format, then accuracy could be specified or measured by the number of spelling mistakes and the number of deviations from the specified format, and completeness by the number of words of the document transcribed divided by the number of words in the source document.

5.3.2. Efficiency

Measures of efficiency relate the level of effectiveness achieved to the expenditure of resources. The resources may be mental or physical effort, which can be used to give measures of human efficiency, or time, which can be used to give a measure of temporal efficiency, or financial cost, which can be used to give a measure of economic efficiency.

From a user's perspective, the time he or she spends carrying out the task, or the effort required to complete the task are the resources he or she consumes. These two types of resources produce two different definitions of efficiency:

$$\textbf{Temporal Efficiency} = \frac{\textbf{Effectiveness}}{\textbf{Task Time}}$$

$$\textbf{Human Efficiency} = \frac{\textbf{Effectiveness}}{\textbf{Effort}}$$

Temporal efficiency can be measured using the MUSiC Performance Measurement Method (section 6.4), while human efficiency can be derived from measures of cognitive workload (section 6.6).

From the point of view of the organisation employing the user, the resource consumed is the cost to the organisation of the user carrying out the task, for instance:

- The labour costs of the user's time
- The cost of the resources and the equipment used
- The cost of any training required by the user

In this case, efficiency can be stated as:

$$\textbf{Economic Efficiency} = \frac{\textbf{Effectiveness}}{\textbf{Total Cost}}$$

For example if the desired goal is to print copies of a report, then efficiency could be specified or measured by the number of usable copies of the report printed, divided by the resources spent on task such as labour hours, process expense and materials consumed.

5.3.3. Satisfaction

Measures of satisfaction describe the perceived usability of the overall system by its users and the acceptability to the system to the people who use it and to other people affected by its use. Measures of satisfaction may relate to specific aspects of the system or may be measures of satisfaction with the overall system.

Satisfaction can be specified and measured by attitude rating scales such as SUMI (section 6.5), but for existing systems attitude can also be assessed indirectly, for instance by measures such as the ratio of positive to negative comments during use, rate of absenteeism, or health problem reports. Measures of satisfaction can provide a useful indication of the user's perception of usability, even if it is not possible to obtain measures of effectiveness and efficiency.

5.3.4. Relative importance of measures

The choice of appropriate measures and level of detail is dependent on which characteristics of the context of use may influence usability and the objectives of the parties involved in the measurement. The importance each measure has relative to the goals of the overall system should be considered. Effectiveness and efficiency are usually a prime concern, but satisfaction may be even more important, for instance where usage is discretionary.

5.4. Derived measures

Some usability objectives will relate to the effectiveness, efficiency and satisfaction of interaction in a particular context. But there are often broader objectives such as learnability or flexibility. These can be assessed by measuring effectiveness, efficiency and satisfaction across a range of contexts.

The learnability of a product can be measured by comparing the quality of use for users over time, or comparing the usability of a product for experienced and inexperienced users. (One of the prerequisites for learnability is that the product implements the dialogue principle "suitability for learning", which refers to the attributes of a product which facilitate learning.)

Flexibility of use by different users for different tasks can be assessed by measuring usability in a number of different contexts. (One contribution to flexibility in use is that the product implements the dialogue principle "suitability for individualisation", which refers to attributes of the product which facilitate adaptation to the user's needs for a given task.)

Usability can also be assessed separately for subsidiary tasks such as maintenance. The maintainability of a product can be assessed by the quality of use of the maintenance procedure: in this case the task is software maintenance, and the user is the software maintainer.

5.5. Item to be designed or evaluated

Although usability depends on the combined characteristics of the components of the context of use, the focus of attention is usually on a specific item which is most often a hardware or software product. However it is also possible to design or evaluate other elements of the context of use such as lighting for the workstation, or a particular type of user. This element becomes the variable which is the object of design or evaluation, while the other elements of the context of use are treated as fixed. Measures of the quality of use of the overall system can be used in this way to compare the appropriateness of alternative versions of elements such as the type of lighting, type of user or amount of user training.

Quality of use measures are not limited to products incorporating software. These type of measures can be used to assess any device with which a user interacts to carry out a task.

5.6. Choosing usability objectives and criteria

Focusing usability objectives on the most important user tasks is likely to be the most practical approach, although it may mean ignoring many functions. Usability objectives may be set at a broad task level (e.g. produce a letter) or a narrow task level (e.g. search and replace) or a feature level (e.g. learnability or adaptability). Setting usability objectives at the broad task level is the most realistic test of usability, but setting objectives at a narrow level may permit evaluation earlier in the development process.

The choice of criterion values of measures of the quality of use depends on the requirements for the overall system and the needs of the organisation setting the criteria. They are often set by comparison with other similar products or systems. When setting criterion values for several users, the criteria may be set as an average (e.g. the average time for completion of a task is no more than 10 minutes), for individuals (e.g. all users can complete the task within 10 minutes), or for a percentage of users (e.g. 90% of users are able to complete the task in 10 minutes). It may be necessary to specify criteria both for the target level of the quality of use and for the minimum acceptable level of the quality of use.

If a task requires several users to collaborate to achieve a goal (group working), usability objectives can be set for both individual and group goals.

6. MUSiC method

6.1. MUSiC project

MUSiC (Metrics for Usability Standards in Computing) is an ESPRIT project which has worked in close conjunction with industry to develop methods and tools for the specification and measurement of usability. Tools have been developed for user-based measures of usability: user performance, user satisfaction, and cognitive workload; and for analytic measures of aspects of usability (SANe). In addition there is a Context Guidelines Handbook to identify the key characteristics of the users, tasks and environments, and an Evaluation Design Manager to guide the choices being made when planning and carrying out an evaluation.

6.2. Context Guidelines Handbook

Usability evaluation and measurement must take place in an appropriate context which matches the context in which the product will be used. A systematic method for describing the context of use and specifying the context of measurement has therefore been developed by the MUSiC project. In cases where it is not feasible to match all of the components of the context of measurement to the context of use, particular care must be taken not to over-generalise from the results of the study. A description of the context of measurement is an essential part of the report of any evaluation.

The Context Guidelines Handbook has the structure shown in Table 1. The Handbook incorporates a Context of Use Questionnaire and a Context Report Table, with practical instructions on how to use them to describe a product's context of use, and to specify an appropriate context of measurement.

The critical components of the context which could affect the usability during an evaluation can be identified and documented. The following procedure (Maissel et al 1991, Macleod et al 1993) is most effective when carried out by a team of people with a stake in the development, support and documentation of the system.

First the team identifies characteristics under each heading, and describes them in fairly broad terms. Next, each characteristic is considered in terms of its potential for affecting usability . (Each decision at this point is based on knowledge of HCI and ergonomics, and experience of any similar product evaluations.) These decisions should be made without concern for the design of any evaluation.

Then it is agreed how each of the items which will or might affect usability will be represented in the context of use for evaluation. The components can either be controlled, monitored or ignored.

Components which are controlled are those characteristics which are fixed or kept within specified ranges, for example, selecting subjects with specific levels of product experience, or carrying out the usability measurement of a product in fixed lighting conditions.

Controlled components may be manipulated to introduce a number of experimental conditions to the evaluation. For example, having two experimental task conditions based on the provision or denial of a help manual. It may be necessary to use a larger number of subjects in each experimental condition in order to draw firm conclusions based on statistical tests.

Components which are monitored are not controlled or selected in any systematic way, but they will be monitored so as to avoid extremes or account for outliers in the data. For example, when subjects are not selected in order to guarantee equal proportions of males to females, but their gender is noted.

If a component is ignored, no attempt is made to control or monitor the value of the component. For example, in an air conditioned office the temperature of the environment might be ignored.

6.3. User-based Performance Measurement

6.3.1. MUSiC Performance Measurement Method

Observing the use of a system gives valuable information about usability, unobtainable by other means. Analysis of what is observed, with the help of a video recording, provides a highly effective means of evaluating usability. To obtain valid and reliable results, the people observed should be representative users performing representative work tasks in appropriate circumstances, and the analysis should be methodical. The MUSiC Performance Measurement Method – developed at the National Physical Laboratory, UK – provides a validated method for deriving performance-based usability metrics. The people observed and the tasks they perform are selected as a result of a context of use study assisted by the MUSiC Context Guidelines Handbook. As part of the analysis, task output must also be assessed. The method is fully documented in the MUSiC Performance Measurement Handbook (Rengger et al., 1993), and is supported by a software tool (DRUM) which greatly speeds up analysis of the video, and helps manage the evaluation.

The Performance Measurement Method gives reliable measures of the effectiveness and efficiency of system use, by evaluating the extent to which specific task goals are achieved, and the times taken to achieve task goals. It also gives measures of time spent unproductively (for example, overcoming problems and seeking help), plus diagnostic data about the location of such difficulties.

These measures enable comparison of prototypes of alternative designs with earlier versions of a system, or with competing products. The diagnostic information helps identify where specific problems are encountered and where improvements need to be made.

6.3.2. Video-assisted usability analysis and DRUM

DRUM, the Diagnostic Recorder for Usability Measurement , is a software tool developed at NPL within the MUSiC Project (Macleod and Rengger, 1993). It supports the MUSiC Performance Measurement Method, and also has wider applicability. Video recording offers considerable advantages for usability evaluation. Video clips of end-users working with a system provide convincing evidence for designers and developers of the usability of their system, and of specific problems. However, analysis is required to convey this information effectively, as an unanalysed interaction log contains too much low-level detail.

Video analysis has previously been very time-consuming, with analysis times of ten hours for every hour of video being typical. It can now be performed much more quickly using DRUM – two or three hours to analyse one hour of video. DRUM supports the management and analysis of usability evaluations, including the derivation of usability metrics, and the identification of evaluator-defined critical incidents for diagnostic evaluation. DRUM assists in many aspects of the evaluator's work:

- management of data through all stages of an evaluation
- task analysis to assist identification and analysis of specific events and usability problems
- video control, and creation of an interaction log of each evaluation session
- automated find and video replay of any logged event
- analysis of logged data and calculation of metrics

DRUM was produced after extensive requirements capture from usability analysts, and study of pre-existing evaluation support tools. It has been iteratively developed since 1990 in collaboration with industry to meet the identified needs of usability testing. DRUM has a graphical user interface, online context-sensitive help and a comprehensive user manual (Macleod et al., 1992). It runs on Apple Macintosh, and drives a variety of video machines.

DRUM's support for task analysis allows evaluators to pre-define, at a suitable level of analysis, the events they wish to log (hierarchically organised if required). This subsequently enables the evaluator to build up time-stamped records of observed events, avoiding difficulties of data analysis which can be encountered with capture of data at the low level of keystrokes and mouse events (Theaker et al., 1989; Hammontree et al., 1992). Initial logging of events can be carried out in real time, if so desired. Most logging is generally carried out retrospectively. Comments can be added to log entries at any time, and entries can be edited. Full control of the video is provided by DRUM, including a variable speed shuttle control. Once any event has been logged, it can be automatically located on the video, and reviewed. DRUM gives easy access to previously created logs and other evaluation data files from its database.

The DRUM metrics processor calculates and delivers usability measures and metrics, which can be inspected and grouped within DRUM, or exported to a statistics package for further analysis.

6.4. Measures of the Task Performance of Users

6.4.1. Task Effectiveness

In the MUSiC Performance Measurement Method the effectiveness with which a user uses a product to carry out a task is comprised of two components: the quantity of the task the user completes, and the quality of the goals the user achieves (Rengger et al 1993). Quantity is a measure of the amount of a task completed by a user. It is defined as the proportion of the task goals represented in the output of the task. Quality is a measure of the degree to which the output achieves the task goals.

As Quantity and Quality are both measured as percentages, Task Effectiveness can be calculated as a percentage value:

Task Effectiveness = 1/100 (Quantity x Quality) %

6.4.2. Efficiency

In the Performance Measurement Method, the temporal efficiency of the user is defined as:

$$\textbf{User Efficiency} = \frac{\textbf{Effectiveness}}{\textbf{Task Time}}$$

Efficiency is measured in a particular Context. The values have little meaning unless there are efficiency benchmarks against which to compare them.

For instance, the efficiency measures can be used to compare the efficiency of:

- Two or more similar products, or versions of a product, when used by the same user groups for the same tasks in the same environments

- Two or more types of users when using the same product for the same tasks in the same environment

- Two or more tasks when carried out by the same users on the same product in the same environment.

In each of these examples the relative efficiency of a test condition can be quoted as a percentage of any of the others.

6.4.3. Productive Period

The MUSiC Performance Measurement Method defines the productive period of a task as the proportion of the time a user spends on the task progressing towards the task goals, irrespective of whether the goals are eventually achieved.

Unproductive periods of the task are periods during which users are seeking help (Help Time), searching hidden structures of the product (Search Time) and overcoming problems (Snag Time). Productive Time is therefore defined as the Task Time remaining after Help, Search, and Snag Times have been removed.

The Productive Period of a user is the Productive Time expressed as a percentage of the Task Time ie.

$$\textbf{PP} = \frac{\textbf{Task Time - Help Time - Search Time - Snag Time x 100\%}}{\textbf{Task Time}}$$

6.4.4. Measures of learning

The rate at which a user learns how to use particular products in specified contexts, can be measured by the rate of increase exhibited by individual metrics when the user repeats evaluation sessions. Alternatively the efficiency of a particular user relative to an expert provides an indication of the position on the learning curve that the user has reached.

The MUSiC Relative User Efficiency metric is defined as the ratio (expressed as a percentage) of the efficiency of any user and the efficiency of an expert user in the same Context.

$$\textbf{Relative User Efficiency} = \frac{\textbf{User Efficiency x 100\%}}{\textbf{Expert Efficiency}}$$

$$\textbf{Relative User Efficiency} = \frac{\textbf{User Effectiveness}}{\textbf{Expert Effectiveness}} \textbf{ x } \frac{\textbf{Expert Task Time}}{\textbf{User Task Time}} \textbf{ x 100\%}$$

6.4.5. Grouped results

When measuring the usability of a product by the MUSiC Performance Measurement Method the results from a group of users with similar characteristics are combined to provide mean usability measures.

Testing of groups of up to 10 users is generally found to be a practical proposition as the cost and time for testing larger groups is often prohibitive. However even with 10 users, if there are major differences in the characteristics of the users, then there will be a large variance associated with the mean values of the metrics. Depending on the purpose of the evaluation, it may be appropriate to investigate the reasons for particularly high or low scores.

6.5. User Satisfaction - SUMI[1]

To measure user satisfaction, and hence assesses user perceived software quality, University College Cork has developed the Software Usability Measurement Inventory (SUMI) as part of the MUSiC project (Kirakowski, Porteous and Corbett, 1992). SUMI is an internationally standardised 50-item questionnaire, available in English, German, Dutch, Spanish and Italian. It takes approximately 10 minutes to complete, and contains questions such as:

- Using this software is frustrating
- Learning how to use new functions is difficult

At least 10 representative users are required to get accurate results with SUMI. The results which SUMI provide are based on an extensive standardisation database built from data on a full range of software products such as word processors, spreadsheets, CAD packages, communications programs etc. SUMI results have been shown to be reliable, and to discriminate between different kinds of software products in a valid manner.

SUMI provides three types of measures: an Overall Assessment, a Usability Profile, and Item Consensual Analysis which gives more detailed information.

6.5.1. Overall Assessment

This is a general global assessment of usability, and it is given by a single numerical figure. The global assessment is useful for setting targets, and for quick comparisons between many products or versions of the same product. Output is given in a standard

[1] For further information contact Jurek Kirakowski and Murray Porteous, University College Cork, Human Factors Research Group, Department of Applied Psychology, Cork, Ireland

format on a scale of 0 to 100 with a mean of 50 and a standard deviation of 10, so that most software products will score somewhere between 40 and 60.

6.5.2. Profile of Perceived Usability

This breaks the Overall Assessment down into 5 sub-scales:

Affect, Efficiency, Helpfulness, Control, and Learnability.

The subscales represent the dimensions with which end users structure their judgement when they assess the usability of software.

The sub-scales were identified, confirmed and validated by factor analysis of large amounts of data collected during the development of SUMI and its predecessors, and by discussion with software engineers, human factors experts, end users, etc. Items which make up these subscales have been drawn from a large sample item pool, and have been selected on the basis of their discriminatory power. Output is on a scale of 0 to 100 as for overall assessment.

6.5.3. Item Consensual Analysis

Item Consensual Analysis lists out those items on which the software being rated was significantly better or worse than the standard of comparison. This gives an indication of specific aspects of the software which people consistently like or dislike. It is thus possible to go back and interview users to find out why they gave these ratings. This gives diagnostic information of potential usability defects in the software.

6.6. Cognitive workload[2]

Cognitive workload relates to the mental effort required to perform tasks. It is a useful diagnostic of situations where users have to expend excessive mental effort to achieve acceptable performance, and is particularly important in safety-critical applications. Adequate usability measures should, therefore, include aspects of mental effort as well as just performance.

Within the MUSiC project, valid and reliable measures of cognitive workload have been developed by Delft University of Technology (Wiethoff et al, 1991). Cognitive workload can be measured either by objective or by subjective means.

Objective measures are relatively independent of personal judgements relating to task complexity and are not directly under the conscious control of the subject. The strength of objective measures is that they are unobtrusive and taken during task performance as well as responding quickly to changes in mental effort.

Subjective measures, on the other hand, are obtained from questionnaires which ask people how difficult they find a task. In this case, measures can only be taken after a

[2] For further information contact Bert Arnold, Edo Houwing and Marion Wiethoff, Technische Universiteit Delft, Kanaalweg 2B, Postbus 5050, 2600 GB Delft, Netherlands

task has been finished, and subjects can consciously control the outcome of the evaluation.

6.6.1. Objective measures

Heart Rate Variability is a measure of mental effort which indicates how much the heart rate varies over time. The variation is influenced by blood pressure regulation, temperature regulation and respiration. In particular it has been shown that the variability related to blood pressure regulation is reduced when people invest mental effort. This variation can be distinguished from the other influences by the distinctive frequency band in which this variation shows itself. Under some circumstances respiration can exert an influence on this frequency band, and so it is also necessary to record respiration. The heart rate variability measure is derived directly from the pulse of the heart, so only the occurrence of heartbeats need to be recorded. Heart rate itself may increase in situations in which people invest more mental effort, but may be also influenced by other factors such as stress, movements, etc.

Heart rate is measured by applying three light, easy to attach, electrodes on the chest, connecting them by special cables to an amplifier, and attaching this to a recording system. The electrodes are non-intrusive - within moments of attaching the electrodes, the user rarely continues to notice their presence. Respiration is measured by a transducer held in place by a band, attached around the chest and abdomen of the user. The bands are easy to attach with tape over the individual's clothes and do not hamper movements of the individual.

6.6.2. Subjective Measures

In addition to objective measures, cheaper and easier to use tools are needed in many instances for both laboratory and field experiments. MUSiC supports the use of two questionnaires: the Subjective Mental Effort Questionnaire (SMEQ) and the Task Load Index (TLX).

The SMEQ was developed at the University of Groningen and Delft University of Technology. It contains just one scale and has been carefully designed in such a way that individuals are supported in rating the amount of effort invested during task performance. The SMEQ has been administered in various laboratory and field studies with high validity and reliability values.

The Task load Index (TLX) is a multi-dimensional rating procedure that provides an overall workload score based on a weighted average of ratings on six subscales. Three subscales relate to the demands imposed on the subjects in terms of:

- the amount of mental and perceptual activity required by the task;
- the amount of physical activity required by the task;
- the time pressure felt due to the task.

A further three subscales relate to the interaction of an individual with the task:

- the individual's perception of the degree of success;

- the degree of effort an individual invested;
- the amount of insecurity, discouragement, irritation and stress.

The TLX, developed by NASA, is an internationally widely used and acknowledged technique.

7. Conclusions

The MUSiC methodology is the first comprehensive approach to the measurement of usability. It reinforces the principles in ISO 9241-11, and provides a practical method for specifying and measuring usability during design.

The MUSiC project recommends that the following steps are taken to incorporate usability into design:

- identify business objectives related to usability
- identify critical success factors for usability
- identify appropriate usability goals for these critical success factors
- include usability goals in the requirements specification
- measure usability during development to ensure that goals are met
- feed back information on usability defects into design

Not all usability measures are required in every situation. The number and type of measures used should depend on the business objectives and the resources available. There are often practical constraints:

- To obtain feedback on usability prior to building a prototype, if the interface is fully specified, use SANe (if cost-effective).
- To obtain the users' perception of the usability of a software product already in use, use SUMI (if cost-effective).
- To evaluate the usability of a product or prototype, use the Evaluation Design Manager and/or Context Guidelines to design a usability study (if cost-effective).
  - To obtain the users' perception of the usability, include SUMI
  - To obtain objective measures of usability and diagnostic feedback, select performance measurement tool(s) according to cost-effectiveness
  - To obtain information about cognitive workload, select cognitive workload tool(s) according to cost-effectiveness

Usability studies may be carried out in a usability laboratory, or the data may be captured in the workplace, with subsequent analysis in the laboratory.

Usability as operationalised by MUSiC encompasses all factors which contribute to the user's view of quality. Other qualities of the product contribute to this global objective.

In this sense usability is the critical quality for successful and cost-effective use of any interactive product.

Acknowledgements

The MUSiC project is part funded by the Commission of the European Communities. The work at the National Physical Laboratory was jointly supported by the UK Department of Trade and Industry. Some of the material in this paper is adapted from Bevan and Macleod (1993). Section 4 of this paper is based on ISO CD 9241-11 (1993) and Macleod et al (1993). Section 5 is based on ISO CD 9241-11 (1993) and Rengger et al (1993).

References

Bevan N (1991a) Enforcement of HCI? Computer Bulletin, May 1991.

Bevan N (1991b) Standards relevant to European Directives for display terminals. In: Bullinger (1991).

Bevan N and Macleod M (1993) Usability assessment and measurement. In Kelly, M (ed), The management and measurement of software quality. Ashgate Technical/Gower Press.

Brooke J, Bevan N, Brigham F, Harker S, Youmans D (1990). Usability statements and standardisation - work in progress in ISO. In: Human Computer Interaction - INTERACT'90, D Diaper et al (ed), Elsevier.

Bullinger HJ (1991) Proceedings of the 4th International Conference on Human Computer Interaction, Stuttgart, September 1991. Elsevier.

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.

CEC (1990) Minimum Safety and Health Requirements for Work With Display Screen Equipment Directive (90/270/EEC) Official Journal of the European Communities No L 156, 21/6/90.

Grudin, J. (1989) The Case Against User Interface Consistency, Comm ACM 32(10), 1164-1173.

Gunsthövel D and Bösser T (1991) Predictive metrics for usability. In: Bullinger (1991).

Hammontree, M.L., et al. Integrated data capture and analysis tools for research and testing on graphical user interfaces, in Proc. CHI'92, ACM Press, pp 431-432.

IBM (1991a) SAA CUA Guide to user interface design. IBM Document Number SC34-4289-00.

IBM (1991b) SAA CUA Advanced interface design. IBM Document number SC34-4290-00.

ISO (1991) ISO 9126: Software product evaluation - Quality characteristics and guidelines for their use.

ISO (1993a) ISO 9241: Ergonomic requirements for office work with visual display terminals (parts 1 to 17).

ISO (1993b) ISO DIS 9241-10: Dialogue principles.

ISO (1993c) ISO CD 9241-11: Guidelines for specifying and measuring usability.

ISO (1993d) ISO DIS 9241-14: Menu dialogues.

Karat C.M., Campbell R., Fiegel T. (1992) Comparison of empirical testing and walkthrough methods in user interface evaluation. CHI'92 Conference Proceedings, Monterey, May 1992. ACM.

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.

Kirakowski J, Porteous M, Corbett M (1992) How to use the software usability measurement inventory: the user's view of software quality. Proceedings of European Conference on Software Quality, 3-6 November 1992, Madrid.

McGinley J and Hunter G (1992) SCOPE catalogue of software quality assessment procedures, 3: Usability section. Verilog, 150 Rue Nicolas-Vauquelin, Toulouse, France.

Macleod M, Drynan A, Blaney M. (1992) DRUM User Guide. National Physical Laboratory, DITC, Teddington, UK.

Macleod M, Thomas C, Dillon A, Maguire M, Sweeney M, Maissel J, Rengger R. (1993) Context guidelines handbook, Version 3. National Physical Laboratory, Teddington, UK.

Macleod M and Rengger R. (1993) The Development of DRUM: A Software Tool for Video-assisted Usability Evaluation. In People and Computers VII, Cambridge University Press.

Maissel J, Dillon A, Maguire M, Rengger R and Sweeney M (1991) Context guidelines handbook. MUSiC Project Deliverable IF2.2.2, National Physical Laboratory, Teddington, UK.

Microsoft (1992) The Windows interface - An application design guide. Microsoft Press, Redmond, USA.

Nielsen J (1993) Usability Engineering. Academic Press.

Nielsen J. and Mack R.L. (Eds.) (1994). Usability Inspection Methods. John Wiley & Sons.

Oppermann R, Murchner B, Paetau M, Pieper M, Simm H, Stellmacher I (1989) Evaluation of dialog systems. GMD, St Augustin, Germany.

Ravden and Johnson (1989) Evaluating the usability of human-computer interfaces. Ellis Horwood, Chichester.

Reiterer H (1992) EVADIS II: A new method to evaluate user interfaces. In People and Computers VII, Monk (ed), Cambridge University Press.

Rengger R, Macleod M, Bowden R, Blaney M, Bevan N. (1993) MUSiC Performance Measurement Handbook. National Physical Laboratory, DITC, Teddington, UK.

Smith S L, Mosier J N (1986) Guidelines for designing user interface software. MITRE Corporation, Bedford, Mass. ESD-TR-86-278.

Theaker, C.J., et al. HIMS: A Tool for HCI Evaluations, in Proc. HCI'89 Conf, (Nottingham, UK, 5-8 Sept 1989) Cambridge University Press, pp 427-439.

Whiteside J, Bennett J, Holzblatt K (1988) Usability engineering: our experience and evolution. In: Handbook of Human-Computer Interaction, M Helander (ed). Elsevier.

Wiethoff M, Arnold AG, Houwing EM (1991) The value of psychophysiological measures in human-computer interaction. In: Bullinger (1991).

A Keystroke-Level Model analysis describes interaction at the level of individual keystrokes and mouse movements. 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. In academic terms, the K-LM is of largely historical interest. since it is not concerned with higher cognitive issues. However, the rapid growth of GUI development has revealed that low level analyses should still be applied. GUIs are being developed which force users to perform cumbersome sequences of mouse actions, to carry out simple operations. Experts may prefer to learn single keystroke alternatives for frequently performed operations. A K-LM analysis of such operations will identify the advantage in actions and time saved.