Software Quality Journal, 4, 115-150 (1995)

Measuring usability as quality of use

Nigel Bevan
NPL Usability Services
National Physical Laboratory
Teddington
Middx
TW11 0LW
UK

tel: +44 181 943 6993
fax: +44 181 943 6306
Nigel.Bevan@ npl.co.uk

Abstract

The conventional assumption that quality is an attribute of a product is misleading, as the attributes required for quality will depend on how the product is used. Quality of use is therefore defined as the extent to which a product satisfies stated and implied needs when used under stated conditions. Quality of use can be used to measure usability as the extent to which specified goals can be achieved with effectiveness, efficiency and satisfaction by specified users carrying out specified tasks in specified environments. Practical and reliable methods of measuring quality of use have been developed by the MUSiC project. These provide criteria for usability which can be incorporated into a quality system. A description is given of the MUSiC methods for specifying the context of use and measuring effectiveness, efficiency, and satisfaction.

1. What is usability?

Considerable confusion exists over the meaning of the term usability. Although the importance of usability as an objective is now widely recognised, it has proved difficult to operationalise usability and integrate it into conventional software engineering practice. This paper distinguishes two complementary but distinctly different approaches to usability. One is a product-oriented "bottom-up" view which identifies usability with ease of use, and the other is a broader "top-down" approach in which usability is interpreted as the ability to use a product for its intended purpose. The product-oriented fits well with conventional software engineering practice, while the broad approach originates from human factors. The contention that a product may be "usable, but not useful" makes sense if usability is defined in terms of ease of use, but is a contradiction in terms from the broader view point.

Usability practitioners frequently complain that work on usability is too little and too late (Bias and Mayhew 1994). This is because usability is often considered only in terms of the ease of use of the user interface. From this perspective, usability is seen as one relatively independent contribution to software quality (as in ISO 9126). In contrast human factors has long argued that usability can only be achieved as the result of a continuous process of user-centred design.

Following the human factors approach, this paper argues that usability as an objective is synonymous with quality of use, ie that the product can be used in the real world. Thus usability has two complementary roles in design: as an attribute which must be designed into a product, and as the highest level quality objective which should be the overall objective of design. As with other quality objectives, it is necessary to set and evaluate measurable targets for usability, and to identify and rectify usability defects. This can be done with the MUSiC methods described in section 4.

2. Approaches to quality

2.1 What is quality?

There is a close analogy between different interpretations of the term usability and comparable interpretations of the term quality. Although the term quality seems self-explanatory in everyday usage, in practice there are many different views of what it means and how it should be achieved as part of a software production process.

Garvin (1984) distinguishes between five overall approaches to defining quality. A traditional view is that quality is transcendent: a simple unanalyzable property which is recognised through experience. Although the term quality often raises this pre-conception, it is an ideal view which does not provide any indication of how quality can be achieved in practice. Garvin distinguishes four other practical approaches to quality:

Product quality: an inherent characteristic of the product determined by the presence or absence of measurable product attributes.

Manufacturing quality: a product which conforms to specified requirements.

User perceived quality: the combination of product attributes which provide the greatest satisfaction to a specified user.

Economic quality: a product which provides performance at an acceptable price, or conformance to requirements at an acceptable cost.

Rather than debate which (if any) of these definitions of quality is correct, they should be recognised as distinctly different approaches, each of which has value for its own purpose.

Quality is generally treated as a property of a product, thus the product view of quality seeks to identify those attributes which can be designed into a product or evaluated to ensure quality. ISO 9126 takes this approach and categorises the attributes of software quality as: functionality, efficiency, usability, reliability, maintainability and portability.

Another approach to quality which has been widely taken up is the use of the ISO 9000 standards to achieve manufacturing quality. ISO 9001 specifies what is required for quality systems. A quality system is a documented set of procedures intended to ensure that a product will meet initially stated requirements. A quality system is a desirable (though not sufficient) condition for achieving quality of the end product. The problem with quality systems has been that it is possible to set up unduly complicated and bureaucratic systems which focus more on the minutiae of the procedures to be followed than on the overall objective of achieving quality of the end product. It is however difficult to achieve product quality in a large project without an effective quality system.

The specification of product quality provides the requirement for the quality system. Manufacturing quality is achieved if the product matches the specification, although the quality of the end product can be no better than the quality of the specification.

Economic quality is a broader approach which takes account of the need to make trade-offs between cost and product quality in the manufacturing process, or price and product quality when purchasing.

ISO 8402 defines quality as:

Quality: the totality of characteristics of an entity that bear on its ability to satisfy stated and implied needs.

This definition is in terms of the characteristics of a product. To the extent that user needs are well-defined and common to the intended users it implies that quality is an inherent attribute of the product. However, if different groups of users have different needs, then they may require different characteristics for a product to have quality, so that assessment of quality becomes dependent on the perception of the user.

2.2 User perceived quality and quality of use

Most approaches to software quality do not deal explicitly with user-perceived quality. User-perceived quality is regarded as an intrinsically inaccurate judgement of product quality. For instance Garvin, 1984, observes that "Perceptions of quality can be as subjective as assessments of aesthetics.".

However, there is a more fundamental reason for being concerned with user-perceived quality. Products can only have quality in relation to their intended purpose. For instance, the quality attributes of a racing car will be very different from a family car. For conventional products this is assumed to be self-evident. For general-purpose products it creates a problem. A text editor could be used by programmers for producing code, or by secretaries for producing letters. Some of the quality attributes required will be the same, but others will be different. Even for a word processor, the functionality, usability and efficiency attributes required by a trained user may be very different from those required by an occasional user.

Work on usability has led to another broader and potentially important view of quality which has been outside the scope of most existing quality systems. This embraces user-perceived quality by relating quality to the needs of the user of an interactive product:

> *Quality of use:* the extent to which a product satisfies stated and implied needs when used under stated conditions.

This moves the focus of quality from the product in isolation to the particular users of the product, the tasks and the context in which it is used. The purpose of a product is to help the user achieve particular goals, which means that measures of quality of use can be defined as:

> *Quality of use measures:* The effectiveness, efficiency and satisfaction with which specified users can achieve specified goals in specified environments.

A product meets the requirements of the user if it is effective (accurate and complete), efficient in use of time and resources, and satisfying, regardless of the specific attributes it possesses.

Specifying requirements in terms of performance has many benefits. This is recognised in the rules for drafting ISO standards (ISO, 1992) which suggest that to provide design flexibility, standards should specify the performance required of a product rather than the technical attributes needed to achieve the performance.

Quality of use is a means of applying this principle to the performance which a product enables a human to achieve. An example is the ISO standard for VDT display screens (ISO 9241-3). The purpose of the standard is to ensure that the screen has the technical attributes required to achieve quality of use. The current version of the standard is specified in terms of the technical attributes of a traditional CRT. It is intended to extend the standard to permit alternative new

technology screens to conform if it can be demonstrated that users are as effective, efficient and satisfied with the new screen as with an existing screen which meets the technical specifications.

2.3 Software quality of use

The same principle can be applied to software. Software quality attributes will determine the quality of use of a software product when it is used in a particular context. Software quality attributes are the cause, quality of use the effect. Quality of use is (or at least should be) the objective, software product quality is the means of achieving it.

Experience has shown that it is almost impossible to accurately specify a set of internal software attributes which will ensure that the requirements for quality of use are met (ie that a given group of users will be able to carry out a specified set of tasks effectively, efficiently and with satisfaction).

The implementation of ISO 9000 quality systems and good software engineering practice are an effective means of ensuring that the end product has the required software quality attributes. However in many approaches to software design, the responsibility of the manufacturer stops at this point. There is often a contractual requirement to deliver a product which matches the specification. Product developers regard it as inevitable that different users will have different perceptions of the quality of the product. As user perceived quality is a subjective judgement outside the control of the developer, meeting the technical specification becomes the sole objective of design.

Some software houses even welcome this situation, as they know from experience that users will rarely be satisfied with a product, even it matches the original specification. This inevitably means an extension of the original contract in order to implement changes to the specification. Even for software systems which are to be used in-house, there is little incentive for the department developing the software to complicate the design and potentially increase costs by asking too closely how users will actually use the software.

One solution is to include target values for quality of use in the requirements specification. ISO 9241-11 explains how this can be done as part of an ISO 9000 quality system, and the MUSiC project has provided practical tools and techniques for specifying and measuring quality of use (see section 4). The use of prototyping and iterative development methods enable prototype versions of the product to be evaluated against a quality of use specification. This is consistent with the approach to usability developed by Whiteside, Bennett and Holzblatt (1988).

2.4 Context of use

The quality of use is determined not only by the product, but also by the context in which it is used: the particular users, tasks and environments. The quality of use (measured as effectiveness, efficiency and satisfaction) is a result of the

interaction between the user and product while carrying out a task in a technical, physical, social and organisational environment (Figure 1).

Measures of quality of use can be used to evaluate the suitability of a product for use in a particular context. However the measures of quality of use also depend on the nature of the user, task and environment - they are a property of the whole "work system" (ISO 1981). Measures of quality of use can thus also be used to assess the suitability of any other component of the context. For instance whether a particular user has the necessary training or skill to operate a product, which tasks a product should be used for, or whether changes in the physical environment (such as improved lighting) improve quality of use.

Similarly the focus of the evaluation (element to be varied) may be a complete computer system, the complete software, a specific software component, or a specific aspect of a software component. Any relevant aspect of software quality may contribute to quality of use, but for interactive software ease of use is often a crucial issue. Quality of use thus provides a means of measuring the usability of a product, and usability is defined in this way in ISO 9241-11.

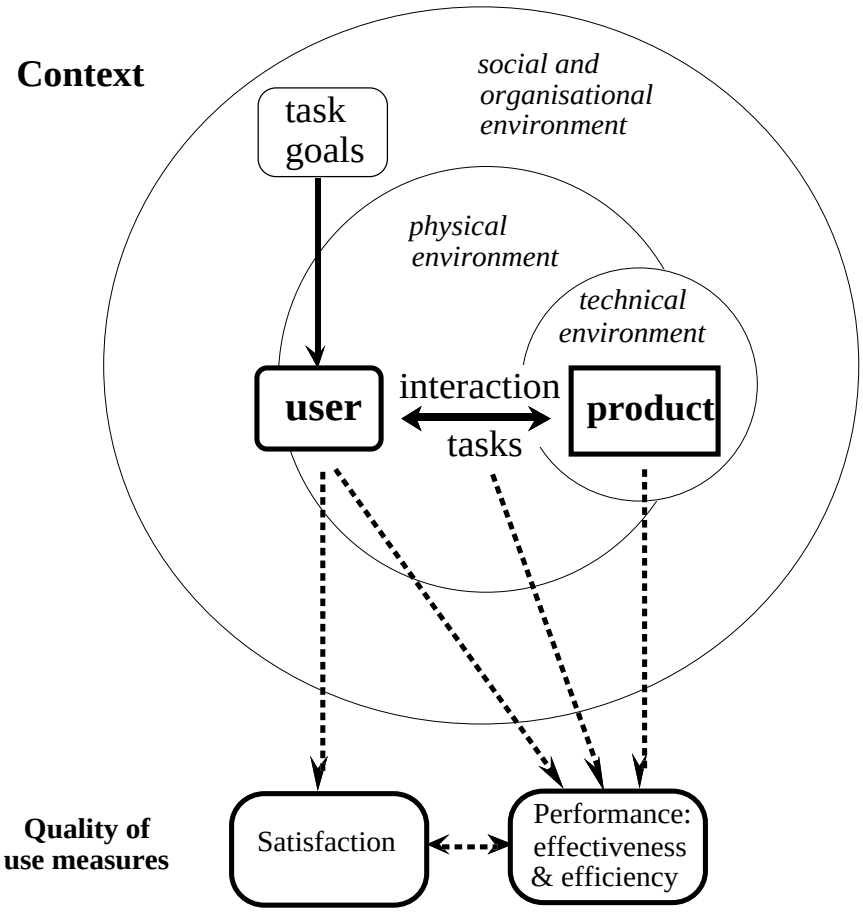

Figure 1 Quality of Use Measures Determined by the Context of Use

3. Measurement of usability

The usability attributes which contribute to quality of use will include the style and properties of the user interface, the dialogue structure, and the nature of the functionality. Measures of quality of use provide the criteria which determine whether the design of the attributes is successful in achieving usability.

It is in principle possible to evaluate usability directly from the usability attributes of the product. This can be done at a number of levels:

• Style guides such as IBM CUA (IBM 1991a, 1991b) or Windows (Microsoft, 1992) can be used. These provide the raw material for an interface, but usability is dependent on the extent to which a dialogue implemented in a particular style is successful in supporting the user's task.

• Detailed attributes of the user interface can be evaluated, for instance using an ISO standard such as ISO-9241-14 for menu interfaces.

• Individual features can be assessed, such as the presence of a help system or the use of a graphical interface. These are examples of functionality which generally contribute to usability. However this type of functionality alone will not guarantee usability, and particular aspects may not be required in every case.

• General usability principles can be used such as the need for consistency, to be self-explanatory and to meet user expectations, such as those in ISO 9241-10. These are examples of useful guidelines for design, but they are difficult to use for evaluation as guidelines are imprecise, not universally applicable and may conflict, and there is no way to weight the relative importance of the individual items for usability in any particular conditions.

There have been several attempts to use checklists as a basis for evaluating usability (eg McGinley and Hunter, 1992; Ravden and Johnson, 1989; and Reiterer, 1992). Usability guidelines and checklists are useful aids for design, and can be used to make quick expert assessments of user interface design, but they do not provide a reliable means of assessing whether a product is usable.

One potential obstacle to a more business and user-centred approach to quality is that measures of quality of use appear to be unreliable and unscientific as they depend on the characteristics of individual users. However this variability is well recognised in the behavioural sciences, and there are well-established techniques for defining the reliability and generalisability of data obtained from human users (eg Kirk, 1968). It is particularly important that the data is obtained from representative users in realistic circumstances. To obtain accurate estimates of quality of use may require a large sample of users to be tested. However the immediate business relevance of the data (can users successfully complete the task? what problems did they have? how long did they take? are they dissatisfied?) means that even the raw data from a small sample of representative users may be highly instructive in identifying the need for changes in design (Nielsen, 1993).

Most current work on usability focuses on techniques to be used to improve the usability of products (eg Nielsen, 1994). Given the choice between using resources for diagnosing usability problems or measuring the quality of use, the value of measurement may not be immediately evident. However, in order to incorporate usability into a quality system, usability requirements need to be specified in measurable terms, and the extent to which these requirements are met needs to be monitored during design. Specifying and measuring usability as quality of use provides several potential benefits:

- unless usability is an objective criterion in the requirements specification, there is often little incentive to put resources into designing for usability;

- measuring against usability objectives provides a means of judging how much further work (if any) on usability is required in order to reach the objectives;

- it provides a means of establishing benchmarks and making comparisons with alternative designs, with earlier versions of a system, or with competing products.

4. MUSiC Methods

The MUSiC methods were specifically developed by the European MUSiC (Metrics for Usability Standards in Computing) project to provide valid and reliable means of specifying and measuring usability, while also giving diagnostic feedback which enables the design to be modified to improve usability. MUSiC includes tools and techniques for measuring user performance and satisfaction.

4.1 Measuring User Performance

The MUSiC 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. The diagnostic information helps identify where specific problems are encountered and where improvements need to be made.

The people observed and the tasks they perform are selected as a result of a context of use study assisted by the MUSiC Usability Context Analysis Handbook. 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.

4.1.1 Context used for evaluation

Unless the evaluation 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.

A systematic method for describing the context of use and specifying the context used for evaluation has been developed by the MUSiC project. In cases where it is not feasible to match all of the components of the context of evaluation 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 evaluation is an essential part of the report of any evaluation.

The MUSiC Usability Context Analysis Handbook has the structure shown in Table I (based on Macleod et al, 1991). 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 for evaluation.

A detailed description of the intended context of use should ideally form part of the requirements specification for the product. As this is rarely the case, it is generally necessary, prior to any evaluation, to hold a context meeting. At the meeting, the relevant stakeholders including the product manager, developers, trainers, usability specialists and user representatives work together to agree on how the product is expected to be used and which user types and tasks should be the focus of the evaluation (Macleod, 1994).

EQUIPMENT	USERS	TASK	ENVIRONMENT
Basic description Product identification Product description Main application areas Major functions **Specification** Hardware Software Materials Other Items	**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** **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 **Technical** **Configuration** Hardware Software Reference materials **Physical** **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

4.1.2 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.

DRUM can be used in real time to collect and store usability evaluation data, and to mark up evaluator-defined critical incidents for diagnostic evaluation. Basic usability metrics associated with task time and the frequency and duration of events can easily be derived and archived, and marked incidents can be accessed automatically. DRUM also supports retrospective analysis which has previously been very time-consuming, with analysis times of up to ten hours for every hour of video. It can now be performed much more quickly using DRUM – two or three hours to analyse one hour of video.

4.1.3 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.

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:

$$\text{Task Effectiveness} = 1/100\ (\text{Quantity} \times \text{Quality})\ \%$$

It is sometimes necessary to calculate effectiveness for a number of sub-tasks, for instance this might be the individual elements in a drawing task. The average effectiveness across sub-tasks is a useful measure of the product's capabilities, but may not reflect the effectiveness of the final task output. For instance, if the user

was unable to save the final drawing, the overall effectiveness would be zero. Similarly, the effectiveness may be reduced if the final drawing contains additional unwanted elements.

4.1.4 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. For example:

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

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.

Task time itself is sometimes used as a usability measure. This is appropriate if all users complete the task satisfactorily (ie with 100% effectiveness). The advantage of the efficiency measure is that it provides a more general measure of work rate by trading off the quantity and quality of output against time.

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

This provides a means of measuring the economic quality. 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.

4.1.5 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.

$$\text{Productive period} = \frac{\text{Task Time - unproductive time}}{\text{Task Time}} \times 100\%$$

Measures of productive period help explain and interpret results for effectiveness and efficiency. They are produced by retrospective analysis of a videotape using DRUM. At the same time valuable diagnostic information is obtained about when specific problems are encountered and where improvements need to be made.

4.1.6 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.

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

4.2 Measuring satisfaction

Satisfaction is composed of comfort and acceptability of use. Comfort refers to overall physiological or emotional responses to use of the system (whether the user feels good, warm, and pleased, or tense and uncomfortable). Acceptability of use may measure overall attitude towards the system, or the user's perception of specific aspects such as whether the user feels that the system supports the way they carry out their tasks, do they feel in command of the system, is the system helpful and easy to learn. If satisfaction is low when efficiency is high, it is likely that the user's goals do not match the goals selected for measurement of efficiency.

Satisfaction can be specified and measured by attitude rating scales such as SUMI (see below), 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.

Cognitive workload is closely related to comfort: even if a system is apparently acceptable for use, it may be low in comfort if it demands too little or too much mental effort. A task demanding too little mental effort may result in a lowered efficiency because it leads to boredom and lack of vigilance, which directly lowers effectiveness. Excessive cognitive workload may also result in lowered effectiveness, if it causes information to be missed and results in errors. This is a particularly important issue in situations where safety is critical, e.g. air traffic control and process control. Measures of cognitive workload can be used to predict these types of problems.

4.2.1 Subjective usability - SUMI

To measure user satisfaction, and hence assess 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 seven languages. 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 an Overall Assessment and a Usability Profile which breaks the Overall Assessment down into 5 sub-scales:

Affect, Efficiency, Helpfulness, Control, and Learnability.

Item Consensual Analysis can be used to list those questionnaire items on which the software being rated was significantly better or worse than the standard of comparison. This provides valuable evidence of specific short-comings of the software.

4.2.2 Cognitive workload

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 (Houwing, Wiethoff and Arnold, 1993).

Subjective measures of cognitive workload can be obtained from questionnaires which ask people how difficult they find a task. 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 (Zijlstra, 1993). 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. The TLX, developed by NASA (NASA-Ames Research Center, 1986), is an internationally widely used and acknowledged technique.

4.3 Case study

This case study briefly summarises some of the results which can be obtained using MUSiC methods. It is a synthesis based on the material gathered during a number of evaluations of customer transactions processing systems for European banking institutions. Many banking organisations are modernising their branch computer systems originally developed during the 1970's. In the case being considered, the objectives for the development of these new systems include:

- staff productivity should increase
- the new system should be perceived to be easier to use than the old system
- the new system should be easy to learn

The specified usability goal was that the new system at roll out should have higher quality of use than the old system. A context meeting was held to plan the evaluation. Some of the important details identified included:

Users: typically bank counter clerks. There are considerably more women than men and the typical age range is 20 - 35. They have a good second level education. For many their only experience with computers is the old transaction processing system.

Tasks: quite variable and reactive in nature as clerks continually respond to the needs of the customer present. It is important that customers are not kept waiting for long periods of time. Depositing and withdrawing cash and cheques are key tasks.

Organisational environment: the bank clerk's role is a well defined activity with a clear set of responsibilities and rules for carrying out procedures.

Technical environment: the old system is mainframe-based using dumb terminals with monochrome screens. The new systems will have colour screens with a Windows interface. Users will require at least 3 hours training in the use of the new system.

The context information enabled a branch environment to be realistically simulated in a special usability laboratory. Representative users were selected to carry out key tasks in response to role-playing customers.

Performance measures

As an example, the efficiency results for depositing and withdrawing cash and cheques with the old and new systems are shown in Figure 2.

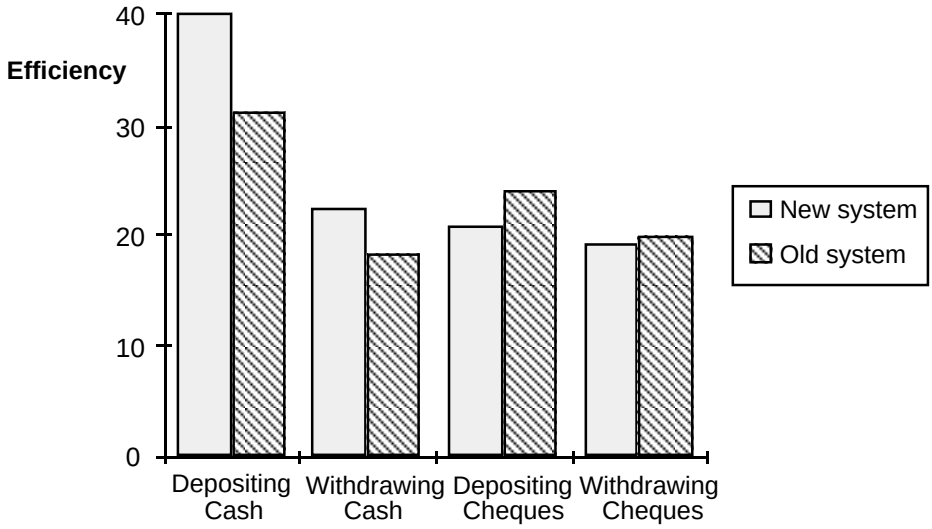

Figure 2. Efficiency (percentage effectiveness per minute)

The new system shows improvements for cash but not cheque transactions. The videotapes were inspected to identify the causes of poorer efficiency for cheque transactions, so that this could be improved in the next prototype in order to meet the initial goals.

Satisfaction - SUMI

SUMI was administered after the users had completed the transaction tasks with each system. The results for the old system shown in Figure 3 reflect its antiquated nature. Although it was quite efficient, it was not well-liked. The results for the new system show that the goal of perceived ease of use has been met. The high rating for learnability is also very encouraging, particularly for a prototype.

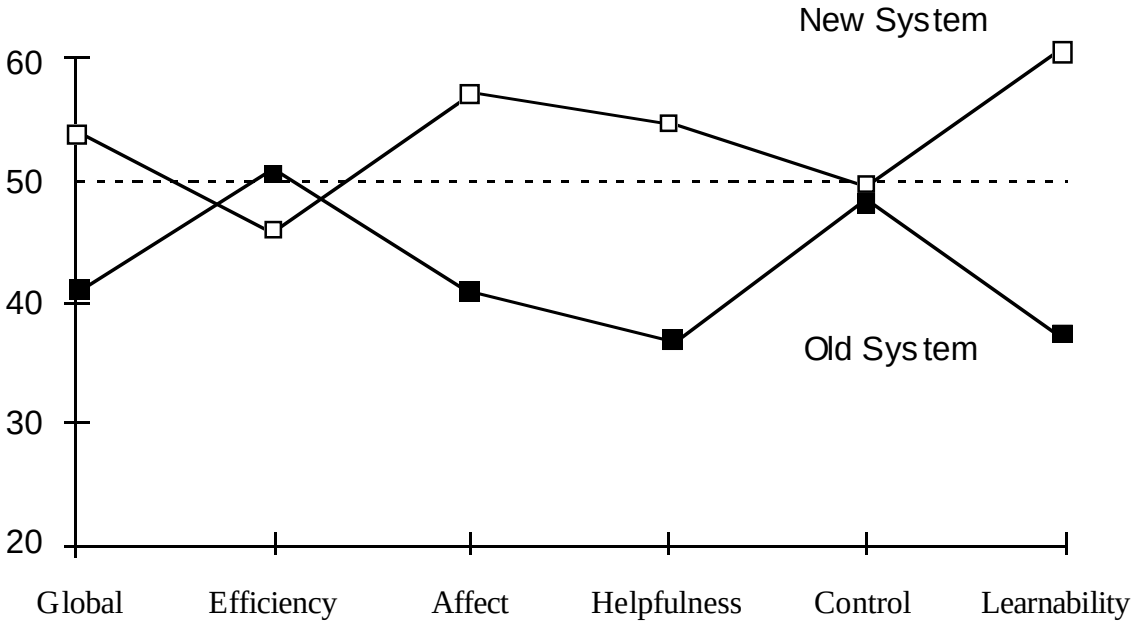

Figure 3. SUMI scores

5. Usability measurement in the design process

The major benefit of using MUSiC methods for usability measurement is to provide a means of specifying goals for usability in terms of quality of use, and a means of evaluating whether they have been achieved. The goals are expressed in terms of the purpose of office products which is to enable users to achieve tasks more effectively, efficiently and with more satisfaction.

The performance measures are thus closely linked to business objectives:

• Whether (or what proportion of) typical users can correctly complete the task (Effectiveness measure).

• The productivity of typical users (Efficiency, measured as Effectiveness/time).

• How the efficiency of typical users compares with an expert user: this gives an indication of the point on the learning curve and may highlight a need to improve the user interface or to provide more training.

• The proportion of the time for which a user is productive: this may also highlight a need to improve the user interface or to provide more training

The measures listed above (particularly the first three) are particularly useful to enable quality of use requirements to be specified as objective values which can subsequently be measured. The first three measures can be obtained without time-consuming analysis, and only if these targets are not met is it necessary to go into more detail to measure unproductive time and obtain diagnostic information.

DRUM can be used with a videotape to calculate the measures, and to mark events which may require subsequent analysis for diagnostic purposes. DRUM also enables other measures (such as counts of events) to be produced.

These provide measures of quality of use in a particular context. It may also be important to measure:

• learnability: how much training or experience is required to use a product effectively
• flexibility: the extent to which a product is usable for different types of users and tasks

Usability evaluation can be incorporated into the design process in the following way:

• When specifying the requirements for a product, use the Usability Context Analysis guide to specify the actual (or intended) context of use, and the target values for quality of use: effectiveness, efficiency, relative user efficiency, productive period (if required) and satisfaction.

• Carry out low cost usability evaluation techniques early in design (eg use low-fidelity prototypes (Muller et al 1993), usability walk through, expert evaluation, (Nielsen, 1994).

• Once a functioning prototype is available test the product with typical users for typical tasks. Use quick co-operative evaluations with 3 or 4 users to identify the major problems (eg Monk et al 1993). Then carry out a more controlled evaluation using the Performance Measurement Method and SUMI to identify remaining problems and to evaluate whether the target values for quality of use have been achieved.

Acknowledgement

The material in this paper draws from discussions in the standards groups developing ISO 9241-11 and ISO 9126-1 and with colleagues at NPL applying MUSiC methods, including Miles Macleod, Rosemary Bowden and Cathy Thomas.

References

Bevan N, Macleod M (1994) *Usability measurement in context.* Behaviour and Information Technology, **13**, 132-145.

Bias R G and Mayhew D. (1994) *Cost-justifying usability.* Boston: Academic Press.

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

Garvin (1984) *What does "product quality" really mean?* Sloane Management Review, Fall 1984.

Houwing E.M., Wiethoff M., and Arnold A.G. (1993). *Introduction to cognitive workload measurement.* Delft University of Technology, Laboratory for Work & Interaction Technology (WIT Lab).

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

ISO (1987) *ISO 9001: Quality systems - Model for quality assurance in design/development, production, installation and servicing.*

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

ISO (1992) *Directives Part 2 - Methodology for the development of international standards.* ISO/IEC, Switzerland.

ISO (1993) *ISO 9241-3 Visual display requirements.*

ISO (1994a) *ISO DIS 9241-10: Dialogue principles.*

ISO (1994b) *ISO DIS 9241-11: Guidance on usability.*

ISO (1994c) *ISO DIS 9241-14: Menu guidelines.*

ISO (1994) *ISO DIS 8402: Quality Vocabulary.*

Kirakowski J & Corbett M, 1993, SUMI: the Software Usability Measurement Inventory, BJEdTech 24.3 210-214

Kirakowski J (1995) The software usability measurement inventory: background and usage. In: P Jordan, B Thomas, & B Weerdmeester, Usability Evaluation in Industry. Taylor & Frances, UK.

Kirk R.E. (1968) *Experimental design: procedures for the behavioural sciences.*

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

Macleod M (1994) Usability in Context: Improving Quality of Use; in G Bradley and HW Hendricks (eds.) *Human Factors in Organizational Design and Management - IV - Proceedings of the I4th International Symposium on Human Factors in Organizational Design and Management,* (Stockholm, Sweden, May 29 - June 1 1994). Amsterdam, Elsevier / North Holland.

Macleod M, Thomas C, Dillon A, Maissel J, Rengger R., Maguire M, Sweeney M, Corcoran R. and Bevan N. (1993) *Usability Context Analysis 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.

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

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

Monk A, Wright P, Haber J and Davenport L (1993) *Improving your human-computer interface.* Prentice Hall.

Muller M J, White E A, Wildman D M (1993) Taxonomy of participatory design practices - A brief practitioner's guide. Communications of the ACM, **36**(4), 26-28.

NASA-Ames Research Center, Human Performance Group (1986) *Collecting NASA Workload Ratings: A Paper-and-Pencil Package.* Moffet Field, CA: NASA-Ames Research Center.

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

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.

Reiterer H and Oppermann R (1993) Evaluation of user interfaces: EVADIS II. Behaviour and Information Technology, **12**(3), 137-148.

Rengger R, Macleod M, Bowden R, Drynan A and Blaney M. (1993) *MUSiC Performance Measurement Handbook.* National Physical Laboratory, DITC, Teddington, UK.

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

Zijlstra, F.R.H. (1993) *Efficiency in Work Behaviour: a Design Approach for Modern Tools.* Delft: Delft University Press.