

Background to SUMI by Jurek Kirakowski

1. Background

1.1 Early studies

A psychological test, wrote Anne Anastasi in her classic book on the subject (Anastasi, 1968), *is essentially an objective and standardised measure of a sample of behaviour*. That psychological tests should be used to evaluate a computer system by measuring the behavioural disposition of end users towards it seems to be an obvious step. Dolotta and colleagues probably coined the phrase *user-perceived quality* in their book of 1976, and in 1978 Dzida, Herda and Itzfeldt reported the first large-scale questionnaire specifically addressed to the problem of devising a rigorous measure of user-perceived quality. They understood that user-perceived quality would turn out to be a multi-dimensional concept, but they made the further assumption that:

each dimension is defined by a set of system properties each of which is mainly associated with this dimension.

Dzida et al produced a seven-factor structure (which was a precursor of the ISO 9241 draft standard, part 10, see ISO, 1991) from their starting point of a sample of 100 system requirements for user-perceived quality. Their paper reports a study with several replications. Most of their seven factor scales stood up quite well to the replications. However, these researchers started from a consideration of system characteristics (such as input format, response time, detail of explanation, etc.) rather than users' expectations of and attitudes to the system being evaluated. Many aspects of Dzida et al's list of software quality features are found in the EVADIS evaluation procedure (Opperman et al, 1988), of which more will be said below. Such lists place considerable importance on the presence or absence of desirable software features which, on the judgement of experts, are considered to contribute to user satisfaction.

In 1983 Bailey and Pearson published their tool for *measuring and analysing computer user satisfaction* (this questionnaire is reported to have been used in a study published by Deese in 1979 so it may be regarded as roughly contemporaneous with Dzida et al's study). Bailey and Pearson argued that

productivity in computer services means both efficiently supplied and effectively utilised data processing outputs... utilization is directly connected to the user community's sense of satisfaction with those services.

In their paper, these authors point to a number of empirical studies which connect organisational performance and the level of satisfaction of the organisation's users with their software systems and give a critique of other early studies on the concept of user satisfaction. Their questionnaire is directed at middle managers within the organisation, to measure their *overall sense of satisfaction with their present computer experiences*. They produced a list of 39 factors, each factor being measured by 4 polar adjective pairs. The factors identified by Bailey and Pearson refer to the way the company uses information technology with such factors as *Top management involvement*, *Organisational competition with the EDP unit* and *Priorities determination* heading the list.

The Bailey and Pearson questionnaire is reported with relatively high reliabilities for most of the factors on the basis of data obtained from a small and homogenous sample pool. Reservations have been voiced about the representativeness of this pool (e.g., Chin, Diehl and Norman, 1988). Independent reliabilities have not, to our knowledge, been reported. Deese in a validity study (1979) reports problems with the length of the questionnaire and whether the questions relate to present conditions within the organisation or an aggregate of past conditions. However, it is clear that the Bailey and Pearson questionnaire is a useful measure of user satisfaction at an organisational level, and as such has been used in a number of subsequent studies by other researchers, even though Bailey and Pearson themselves modestly admitted that further research was needed in the development of the measurement tool they had proposed.

A paper by Ives, Olson & Baroudi in 1983 reported another measure of user information satisfaction. A short form of this measure was developed later by Baroudi and Orlikowski (1988). This latter is a short 13-item scale consisting of such items as *Relationship with EDP staff*, *Processing of requests for changes* and *Degree of EDP training provided*. It is therefore also a scale aimed more directly at the total informational system a company has installed, perhaps more suitable to an age of mainframes and consoles rather than the way computing resources are distributed in a modern office with networks of workstations. Igarria and Nachman in an independent study (1990) report a high reliability for this scale of 0.89. This is a high value, and it is noteworthy because it was generated independently of the original scale.

1.2 The emergence of the dimensions of usability

The discussion above suggests two approaches towards the development of attempts to measure user satisfaction or quality of use. On one side was an approach which began from a more or less a priori assessment of what can be considered good for an interface, and then evaluated the extent to which a system exhibits these properties; on the other was an approach which began from a consideration of end users' reactions to the system that was being evaluated.

Both approaches had their strong points, but the weak points were that one approach depended on expert's constructs and did not take into account how end users saw matters, and the other had a general focus that did not help much with the evaluation of specific technical details about the interface.

It is worthwhile here to note in passing that a number of scales relating to general attitudes towards computers were developed during this period, which instruct users to rate their attitudes to computers and information technology (see for instance Igarria and Parasuraman, 1991 for a good summary and critique). General scales are useful for the insight they give us into attitudes society has towards computers, and the attitudes society has on the impact of computers in that society. They do not enable the evaluator to focus on specific software systems and will not be considered further in this paper.

The next stage of development of research on quality of use was the development of a number of short scales that simply attempted to measure the end users' degree of satisfaction. Doll and Torkzadeh reported a measure of end-user computing satisfaction in 1988. This is a ten-item measure of the users' reactions to a specific computer interface. The reported reliability of this scale is 0.76, which is not encouraging.

At around this time, a questionnaire for assessing user satisfaction was also circulated via electronic mail originating from Dr J Brooke, of DEC. The scale is called SUS. For commercial confidentiality reasons he did not publish any information regarding the scale's validity or reliability. However, in an independent study in the HFRG laboratories Lucey (1991) demonstrated that this short ten-item scale has a reliability of 0.85. Lucey's figure may be an under-estimate because she reports data from only 77 users. The Doll and Torkzadeh and the SUS questionnaires are interesting because they attempt to capture a user's attitudes to one single interface, thus starting the blend of the two approaches discussed in section 1.1. Neither questionnaire went as far as to attempt to discover any of the latent structure beneath the surface of the concept of satisfaction.

The questionnaire of Lewis (1991) may be considered to represent the apogee of this development. Lewis developed a three-item questionnaire called the After Scenario Questionnaire (ASQ). The ASQ was developed to be used immediately following scenario completion in scenario-based usability studies, where a scenario, according to the author, is a collection of related tasks. ASQ shows impressive reliabilities although the number of users on which the psychometric properties are estimated are small. The three questions of ASQ unequivocally measure one single underlying aspect of users' perceptions of how easily and quickly the scenarios were completed and the contribution of support information to carrying out the tasks. Lewis agrees that more work needs to be done to demonstrate the general usefulness of this questionnaire, and a longer recent questionnaire is also available from Lewis.

Work on specific questionnaire methods of analysing user reactions was started in the Human Factors Research Group (HFRG), University College Cork, in 1986. The direction taken was the same as the

studies of Doll and Torkzadeh and Brooke cited above. That is, the focus of the research was to examine specific user reactions to a specific computer product with which the user sample had some experience.

The first result from the HFRG studies was the Computer User Satisfaction Inventory (CUSI) (see Kirakowski, 1987, Kirakowski and Corbett, 1988). CUSI is a short questionnaire of 22 items. Two subscales of usability were established, called at the time Affect (the degree to which users like the computer system) and Competence (the degree to which users feel supported by the computer system). These subscales were arrived at through cluster analysis of intercorrelations of responses to individual questions in a large initial item pool. This item pool was gathered from literature searching and discussion with end-users about their reactions when carrying out their normal tasks on their usual system. The range of systems sampled was large and heterogeneous.

Initial estimates of the reliability of the CUSI questionnaire on a sample of data independent of the sample on which the original dimensions had been discovered showed an overall reliability of 0.94, with two separate scales showing reliabilities of 0.91 for Affect, and 0.88 for Competence. At about the same time that CUSI was published, Shneiderman included the QUIS scale in his book in 1987. An important evaluation of QUIS was published by Chien, Diehl and Norman in 1988, by which time the questionnaire had been incremented up to version 5.0. Currently the questionnaire may be found in the second edition of Shneiderman's book (1992).

QUIS version 5.0 consists of one introductory section, which is an *overall reactions to the software* scale, and four other sections, each consisting of between 4 to 6 items. These sections are: *Screen, Terminology and System Information, Learning, and System capabilities*. Chin, Diehl and Norman report a high reliability coefficient of 0.94 for the entire scale. The items within the sections are fairly specific to each section's theme and we would therefore expect there to be strong inter-correlations between items within the sections (for instance, in the *Learning* section we find items such as *Learning to operate the system, Exploring new features by trial and error, and Remembering names and use of commands*.) Chin, Diehl and Norman report some problems with the stability of the items within the sections to do with *Screen, and Terminology and System Information*. That is, it is not clear that the four sections of QUIS version 5.0 do actually measure different aspects of end users' experience.

The current version reported by Shneiderman has two forms: a long one and a short one, and a space for users to write in their comments about the system being evaluated. Statistical reliability, cross-correlations, and benchmarking have not, to our knowledge, been achieved or independently assessed for the current versions of QUIS. Two studies carried out in 1990 - 1991 examined the three questionnaires, CUSI, SUS and QUIS version 5.0 and paved the way to the SUMI questionnaire. Lucey (1991) showed that the Affect scale of CUSI correlated well with overall QUIS and SUS. The Competence scale, however, was marked by low correlations with these other questionnaires, and looked at least bi-modal. CUSI Competence therefore looked as if it was measuring some hitherto undiscovered dimension of end-user experience, but that it was not doing it very efficiently.

Wong and Rengger (1990) calculated correlation coefficients between CUSI Affect, CUSI Competence, SUS, and the overall QUIS score. They found that CUSI Affect, SUS, and QUIS all correlated together quite well with coefficients of between 0.672 to 0.744. CUSI Competence however correlated less strongly with these other scales: 0.584 with SUS and 0.379 with QUIS. The authors concluded that CUSI Affect, QUIS and SUS most probably were measuring a dimension which could be called Affect, whereas CUSI Competence, with its low correlations with all the other tests may well be measuring something else.

If you wanted to compare the relative user satisfaction of one system against another with CUSI, as with all the other satisfaction questionnaires reviewed here, you could only do it in a comparative manner, that is, system X against system Y. There were no absolute benchmarks. This was the state of the art at the time when development on SUMI began. However, before we go on to SUMI itself, we should review concurrent developments in checklists and surveys during this period.

2. Checklists and Surveys

In this section, we comment briefly on two kinds of instruments which bear a close resemblance to questionnaires, but which in reality are something else altogether. First survey questionnaires will be considered and then checklist approaches.

2.1 Survey-type questionnaires

We may distinguish between specific attitude scales which are applicable to a broad range of software systems (such as discussed in section 1 above), and bespoke survey-type questionnaires which request users to consider particular aspects of a system which are unique to one or possibly a small family of systems. Such survey-type questionnaires are not problematic when the questions seek information about things which are very clearly not a matter of individual interpretation, and indeed may be extremely helpful in finding out, for instance, patterns of usage. Preece et al (1994) give a good example of a straightforward survey questionnaire. However, when the questions begin to shade into matters which can be interpreted according to the individual judgement of the respondent, the questionnaire begins to take the form of an attitude questionnaire. All attitude questionnaires require several iterations of the cycle of item analysis, data gathering, and statistical analysis before they can be used with confidence. A good recent example of a thoughtfully-developed questionnaire which is part-way between an attitude and a survey questionnaire is reported by Spenklink et al. (1993). Their DES measures one specific aspect of a computer system -- the visual clarity of its displays -- and demonstrates high reliabilities for some of its scales over 3 independent studies. However, in general, the literature has many examples of one-cycle questionnaires, developed as the need arose, and usually characterised by low reliabilities, when these are actually cited. Very rarely are the statistical properties assessed with recourse to a second, independent sample. It would be invidious to point out any particular examples. Suffice to say that in general, standards of reviewing articles for publication are lax in many human factors journals when it comes to questionnaires. In addition, when one looks at the practices of human factors departments in the software industry we find in general an equally gloomy picture. As Marc Chignell comments:

questionnaires are probably the most frequently used method of summative evaluation of user interfaces. However, questionnaires provide a subjective evaluation of interfaces which is often greatly influenced by the type of questions asked and the way in which the questions are phrased. (Chignell, 1990)

--which is precisely why a questionnaire in order to be useful, must be developed with care. Saunders and Jones (1992) reported that 80% of companies they surveyed used some kind of questionnaire to measure user attitudes. Our personal experience would suggest that many of these instruments are simply unknown quantities in terms of their reliabilities, possessing at most a spurious *face validity*: that is, they look good on the outside only. It is ridiculous to consider that a company may well spend literally millions of ECUs developing a system, and leave the end-user evaluation of this large investment to a weak and untested component of the development process. As Thimbleby (1990) asks: why are so many commercial software products faulty? Because users simply don't get a proper chance to express their opinions.

2.2 Checklist approaches

A checklist is a list of things that are considered desirable to be present or to be carried out in order to achieve a certain effect or goal. In an increasingly technological society, especially when there is a high premium to be paid for operator error, checklists are extremely important. In HCI, checklists, with their high technology-related content, are designed for use by experts.

A checklist can be used -- informally -- as an evaluation tool on the basis that an assessment is made of the number of features or elements which are considered to be desirable but which are not present. Guidelines can also be made into checklists, which can also be then used as evaluation tools. Ravden and Johnston (1989) presented a useful checklist for evaluating the usability of human-computer interfaces, for

identifying problem areas... extracting information on problems, difficulties, weaknesses, areas for improvement and so on.

The checklist is designed for use by end users. It is fair to say that end users find the questionnaire daunting, consisting as it does of 15 pages of detailed technical questions. The questions are assembled from a variety of sources clustered into nine criteria (such as *Visual Clarity*, *Consistency* and *Compatibility*). Although the nine criteria sound intuitively plausible no rationale is given for their adoption, or for clustering questions under each criterion. Not all questions will be applicable to every system, and some of the more recent developments in GUI technology are not catered for (not surprisingly, of course given the age of the document). The authors do however urge caution when adding or deleting items. The argument used by Ravden and Johnson is that this method was used and found to be applicable in practice and that the arrangement of information in the questionnaire enabled users to understand and make use of the material and that

although the details of the many guidelines within the literature vary, the underlying principles, or criteria, for 'good' user interface design on which they are based, are generally in agreement.

The authors properly warn of the danger of using their checklist in a simple, summative manner: a temptation which the evaluator should resist strenuously. The problem is of course that not all items in a checklist are of equal importance. The technical orientation of the Ravden and Johnson checklist suggests that it is most useful as an aid to expert-based evaluation. The most prominent of attempts to provide a checklist approach to the evaluation of a computer system is presented in EVADIS (Opperman et al, 1988, 1992). EVADIS II is published only in German, but the English speaking reader may be referred to Reiterer and Opperman (1993) who place the method in a more general framework of user interface evaluation.

As the authors remark, *an expert with a grounding in human factors is needed*. EVADIS II presents a formidable collection of factual questionnaires, checklists, and guidelines. At the core of EVADIS II, we find a list of technical system component test items. The test items in Appendix D of Opperman et al (1992) are extremely similar in wording and content to a number of other extant checklists which are cited in their 1993 paper.

The test items are embodied in a two-dimensional framework of technical system components against ergonomic criteria (based on ISO 9241 part 10 with four additional criteria added by the authors themselves). The evaluator has to give a rating which compares the analysed ergonomic quality with the best attainable ergonomic quality. Software support later averages the ratings given for each ergonomic criterion. As Reiterer and Oppermann comment *the final report can be biased, to a certain degree, by the judgement of the expert with respect to the relevance and rating of the evaluation items*. To which we may add, that averaging item ratings in the described manner is not justified in terms of measurement theory since there is no scaling involved to ensure that the evaluation items are of equal importance. At any rate, Reiterer and Oppermann conclude in 1993 that *no experimental tests are available which demonstrate the validity and the reliability of EVADIS II*.

One is reminded of Nielsen's observation with regard to what he calls heuristic evaluation that *between three to five evaluators seem necessary to find a high proportion of the usability problems* and that *major usability problems have a higher probability than minor problems of being found in a heuristic evaluation* (Nielsen, 1992). Heuristic evaluation is less directive than the approaches outlined above and stems from the work of Molich and Nielsen (1990) who derived nine major heuristics or principles from an analysis of usability problems. They claim that *almost all usability problems fit well into one of the categories*. No empirical evidence for the reliability of this claim has yet been presented. Dzida et al (1993) remark that

although the nine principles seem to be 'obvious', they are difficult to apply in practice. If an evaluator identifies a usability problem it might be hard to express it precisely and allocate it to one of the nine principles.

Molich and Nielsen's approach is aimed directly at the expert evaluator (or, more properly, team of expert evaluators). It may also be said that if there is a major usability problem with a piece of software, this will become evident when the first few end users try the software anyway, so why disturb the experts?

Holcomb and Tharp (1991) presented an interesting model that was intended to aid designers in making initial usability decisions, and to provide a tool for evaluating a particular user interface and assigning it a usability rating. Based on a technical paper by Gould entitled *How to design usable systems* they presented a model with seven major categories of features, and 20 usability attributes each allocated to one of the seven categories. The procedures for selecting these 20 attributes and for allocating them to the seven major categories is not described, and their reliance on Gould is not justified.

However, Holcomb and Tharp present results from an impressively large database in which respondents were asked to judge how the attributes affected usability and how a particular information technology product (WordPerfect 4.2) adhered to the attributes. Analysis of the mismatches between these two sets of judgements enabled the authors to draw a profile of the usability strengths and weaknesses of the product being evaluated. No inferential statistics or even measures of dispersion are given for the comparison of attributes (only one *t* distribution test result is cited in the entire paper) so we cannot therefore make any estimates of the reliability of the method or the number of respondents needed to make an accurate judgement, but their paper does argue persuasively for the overall validity of this kind of approach.

Holcomb and Tharp conclude, and we concur with their general point:

Producers of other widely-used commercial products, for example those in the food industry, do much consumer testing and test marketing for new and improved products before they are released for widespread use. For software to become more usable, and for the use of computers to advance as rapidly as it might, the same type of user feedback needs to be encouraged from the consumers of software products.

3. SUMI Development

Work on SUMI started in late 1990. One of the work packages entrusted to the HFRG within the MUSiC project was to develop questionnaire methods of assessing usability. The objectives of this work package were:

1. to examine the CUSI Competence scale and to expand it and to extract further subscales if warranted by the evidence;
2. to achieve an international standardisation database for the new questionnaire and to validate its use in commercial environments.

Both these objectives were achieved by the end of the project. The SUMI questionnaire was first published in 1993 and has been widely disseminated since then, both in Europe and in the United States.

3.1 Psychometric development

SUMI started with an initial item pool of over 150 items, assembled from previously reported studies (including many reviewed above), from discussions with actual end users about their experiences with information technology, and from suggestions given by experts in HCI and software engineers working in the MUSiC project. The items were examined for consistency of perceived meaning by getting 10 subject matter experts to allocate each item to content areas. Items were then rewritten or eliminated if they produced inconsistent allocations.

The questionnaire developers opted for a Lickert scaling approach, both for historical reasons (the CUSI questionnaire was Lickert-scaled) and because this is considered to be a natural way of eliciting opinions about a software product. Different types of scales in use in questionnaire design within HCI are discussed in Kirakowski and Corbett (1990). The implication is that each item is considered to have roughly similar importance, and that the strength of a user's opinion can be estimated by summing or averaging the individual ratings of strength of opinion for each item. Many items are used in order to overcome variability due to extraneous or irrelevant factors.

This procedure produced the first questionnaire form, which consisted of 75 satisfactory items. The respondents had to decide whether they agreed strongly, agreed, didn't know, disagreed, or disagreed strongly with each of the 75 items in relation to the software they were evaluating.

Questionnaires were administered to 139 end users from a range of organisations (this was called sample 1). The respondents completed the inventory at their work place with the software they were evaluating near at hand. All these respondents were genuine end users who were using the software to accomplish task goals within their organisations for their daily work. The resulting matrix of inter-correlations between items was factor analysed and the items were observed to relate to a number of different meaningful areas of user perception of usability. Five to six groupings emerged which gave acceptable measures of internal consistency and score distributions.

Revisions were made to some items to centralise means and improve item variances, and then the ten best items with highest factor loadings were retained for each grouping. The number of groups of items was set to five. Items were revised in the light of critique from the industrial partners of MUSiC in order to reflect the growing trend towards Graphical User Interfaces. A number of users had remarked that it was difficult to make a judgement over five categories of response for some items. After some discussion, it was decided to change the response categories to three: **Agree**, **Don't Know**, and **Disagree**.

This produced the second questionnaire form of 50 items, in which each subscale was represented by 10 different items.

Typical items from this version are:

Item No. Item Wording

- | | |
|-----|---|
| 1. | This software responds too slowly to inputs. |
| 3. | The instructions and prompts are helpful. |
| 13. | The way that system information is presented is clear and understandable. |
| 22. | I would not like to use this software every day. |

A new sample (sample 2) of data from 143 users in a commercial environment was collected. Analysis of this sample of data showed that item response rates, scale reliabilities, and item-scale correlations were similar to or better than those in the first form's sample. Analyses of variance showed that the questionnaire differentiated between different software systems in the sample. After analysis of sample 2, a few items were revised slightly to improve their scale properties. The subscales were given descriptive labels by the questionnaire developers. These were:

- Efficiency
- Affect
- Helpfulness
- Control
- Learnability.

The precise meaning of these subscales is given in the SUMI manual, but in general, the Affect subscale measures (as before, in CUSI) the user's general emotional reaction to the software -- it may be glossed as Likeability. Efficiency measures the degree to which users feel that the software assists them in their work and is related to the concept of transparency. Helpfulness measures the degree to which the software is self-explanatory, as well as more specific things like the adequacy of help facilities and documentation. The Control dimensions measures the extent to which the user feels in control of the software, as opposed to being controlled by the software, when carrying out the task. Learnability, finally, measures the speed and facility with which the user feels that they have been able to master the system, or to learn how to use new features when necessary.

At this time, a validity study was carried out in a company who has requested to remain anonymous. In this company, two versions of the same editing software were installed for the use of the programmer teams. The users carried out the same kinds of tasks in the same environments, but each programmer team used only one of software versions for most of the time. There were 20 users for version 1, and 22 for

version 2. Analysis of variance showed a significant effect for the SUMI scales, for the difference between systems, and for the interaction between SUMI scales and systems. This last finding was important as it indicated that the SUMI scales were not responding *en masse* but that they were discriminating between differential levels of components of usability. Table 1 shows the means and standard deviations of the SUMI scales and the two software versions.

Table 1: SUMI data from preliminary validity study

| Scale | Version 1 | | Version 2 | |
|--------------|-----------|------|-----------|------|
| | Mean | sd | Mean | sd |
| Efficiency | 21.6 | 3.98 | 24.4 | 3.55 |
| Affect | 19.7 | 3.99 | 25.0 | 2.78 |
| Helpfulness | 21.9 | 4.72 | 23.8 | 4.56 |
| Control | 21.4 | 4.50 | 22.1 | 3.64 |
| Learnability | 23.7 | 4.16 | 23.9 | 4.26 |
| | | n=20 | | n=22 |

In fact, Version 2 was considerably more popular among its users, and the users of this version considered that they were able to carry out their tasks more efficiently with it. Learnability was not considered to be an issue by either group of users. The Data Processing manager of the company reviewed the results with the questionnaire development team and in his opinion, the results confirmed informal feedback and observation. Later we learnt that the company subsequently decided to switch to Version 2, not only on the basis of our results, but because *that was the general feeling*.

At this stage, the Global scale was also derived. The Global scale consists of 25 items out of the 50 which loaded most heavily on a *general usability* factor. Because of the larger number of items which contribute to the Global scale, reliabilities are correspondingly higher. The Global scale was produced in order to represent the single construct of *perceived quality of use* better than a simple average of all the items of the questionnaire.

The first normative sample (sample 3) was then obtained by administering the questionnaire to a large sample of everyday end users of a large range of mainly office-type software systems. In addition to word processors, spreadsheets, database retrieval systems and financial packages, there were also CAD, communications packages, text editors and some programming environments.

Overall there were more than 150 systems evaluated, with over 1,000 users records. An amount of re-analysis was carried out on sample 3. First, reliabilities were estimated.

Table 2: Reliabilities of samples 2 and 3

| Sample: | 2 | 3 |
|--------------|-------|--------|
| Global | 0.90 | 0.92 |
| Efficiency | 0.77 | 0.81 |
| Affect | 0.80 | 0.85 |
| Helpfulness | 0.80 | 0.83 |
| Control | 0.65 | 0.71 |
| Learnability | 0.77 | 0.82 |
| | n=143 | n=1100 |

Using Cronbach's Alpha coefficient.

We can see from table 2 that the questionnaire version employed for sample 3 was generally more reliable. This has now become the standard version of the questionnaire. The individual subscale values are high enough for scales of 10 items and 3 response categories. One could increase the reliabilities by either adding more items to the subscales (and therefore increasing the overall size of the questionnaire), or by increasing the number of response categories. Neither of these two options were considered

appropriate. The sample 3 data is given in the SUMI Handbook as the base normative data, and the selection of types of system evaluated was also taken as a basic set of objects to form a *usability index* for subsequent standardisations.

Although samples 2 and 3 are not strictly speaking comparable because of minor changes in the questionnaire, a re-analysis of the underlying factors was carried out, comparing the first five rotated factors from sample 2 and sample 3. The factor loadings on each item were correlated using Spearman's rho, and the correlation matrix shown in table 3 was obtained:

Table 3: Correlation matrix between factor loadings on samples 2 & 3

| Sample Three Factors: | 1 | 2 | 3 | 4 | 5 | |
|-----------------------|---|------|------|------|------|------|
| Sample | 1 | -.24 | .70 | .45 | .33 | -.14 |
| Two | 2 | .05 | .47 | .71 | .32 | -.41 |
| Factors: | 3 | .78 | -.19 | -.06 | -.16 | -.15 |
| | 4 | .10 | .51 | .14 | .40 | -.15 |
| | 5 | -.58 | .21 | .10 | .20 | .41 |

It will be seen from table 3 that apart from a re-shuffle of the order in which the first three factors emerged from the analysis of sample 3 (which reflects differences about areas of concern between the users of the two samples rather than any divergence in factorial structure) the factor loadings are remarkably consistent. As will be noticed from the reliability estimates in table 2, factor 4 (Control) is the weakest scale. From table 3 it would appear that Control and Efficiency are closely related, and future developments of the SUMI questionnaire should examine the Control dimension in greater detail.

In April 1994, a second standardisation dataset was produced (sample 4), on the same principles as sample 3. This reflects the commitment of the development team to update the standardisation dataset at least once a year to keep track of new products on the market and increasingly strident demands from users for quality of use. Reliabilities have stayed stable between samples 3 and 4.

At this stage it was decided that the psychometric development of SUMI had reached a plateau. After this, the next step is to go back to first principles and start to design a different questionnaire.

3.2 Tool development

The literature on quality of use questionnaires has many questionnaires, but few are developed to the status of a ready-to-use tool, much less an industry-tested one. Three developments took place which brought SUMI to the status of a tool fit for use in an industrial context.

3.2.1 Scale standardisation

The first development was quite simple, and this was to develop a series of formulae which convert the raw scores given by the respondents to a scale whose mean is 50 and standard deviation is 10, on the basis of the information gained in the standardisation sample. Thus it was no longer necessary to do comparative evaluations with SUMI, since the comparisons were already built into the questionnaire. The properties of the standard normal distribution indicate that over 68% of software will find a score on all the SUMI scales within one standard deviation of the mean, that is, between 40 and 60. Software which is above (or below) these points is already, by definition, well above (or below) average.

In fact, each SUMI scale tends to have a longer distribution *tail* at the low end than at the high end. This may well be a property of the usability of many commercially available systems taken together. Many systems will exhibit fairly high levels of usability on one or more scales; very few (only the acknowledged market leaders) will achieve really high scores. However, there will always be a long tail-off of systems in use whose quality of use characteristics are mediocre to poor, either because market forces (other than those driven by quality considerations) keep them in play, or because they occupy specialised market niches in which the competition for quality of use products is not so strong. What this

means for SUMI is that if a product does exhibit some poor quality of use features, these will stand out immediately, because the product will be placed on that part of the standardisation curve where it is hardest to ascend.

3.2.2 Item Consensual Analysis

One of the first developments suggested by the industrial partners of the project was that as much as possible, the MUSiC tools should diagnose usability problems as well as measure usability status. With SUMI this led to the development of the Item Consensual Analysis feature. Item Consensual Analysis (ICA) enables the evaluator to pinpoint more precisely where particular usability problems may lie than can be gained from an examination of the subscale profile.

The standardisation database is used to generate an expected pattern of response for each SUMI item (i.e., how many Agrees, Don't Knows, and Disagrees there should be for each particular item). The expected pattern of response is then compared with the actual, obtained pattern (i.e., how many Agrees, Don't Knows, and Disagrees there actually are for that item). The observed and the expected patterns are then compared using a statistic distributed as Chi Square.

Items on which there is a large discrepancy between expected and obtained patterns of response represent aspects of the system which are unique to the system being evaluated. These may be positive or negative comments on the system. Positive comments indicate the areas in which the system being evaluated may have a market advantage; negative comments indicate the areas in which more work needs to be done or at least where there is scope for improvement.

ICA can only be carried out if there are at least 10 users who have responded to the questionnaire. This is because with smaller sample sizes, the distribution statistic becomes approximate. For some purposes ICA is sufficient, and the evaluator can, together with a knowledge of the software being evaluated, pinpoint specific usability problems simply by examining the six or so largest ICA results and relating them back to what they know about the system.

If the objective of the evaluation is to obtain a more detailed diagnosis of the software features of the product, Item Consensual Analysis may be used within a two-stage sampling approach. In the first stage of sampling a large sample (at least more than 12 users) provides an initial SUMI analysis. The ICA outputs are used to construct an interview schedule, and in the second stage, selected users are interviewed in order to find out specific causes in the software for the extreme user reactions which emerged in the ICA analysis.

In addition to the subscales, and ICA results, there is also one other output from SUMI which facilitates two-stage sampling. This is the individual user profiles. Each user's SUMI Global and subscale scores may be computed individually, and the use of appropriate statistics enables the evaluator to select out users for in-depth interviewing. The SUMI items which come up from ICA act as a series of memory probes for the end users in the evaluation.

It must be pointed out that users will talk about the system in terms that they are familiar with, that is, with the effect of the system on the way they carry out their work. This is the sort of language used by the SUMI questionnaire, which is why it can be used by non-technical end users. The actual diagnosis of these problems and their fix, however, is in the domain of the human factors expert and the software engineer. Users usually cannot and should not be required to suggest a technical fix for a usability problem (see Grudin, 1991, 1992).

The theoretical basis behind the concept of two-stage sampling with ICA involves the adaptation of Flanagan's Critical Incident Technique (Flanagan, 1954) together with the well established phenomenon that cued recall is superior straight recall, as well as the theory of specific memory encoding (Humphreys, Pike, et al, 1988). ICA is an innovative feature of questionnaire technology developed specifically for SUMI, but it is solidly grounded both theoretically and statistically.

3.2.3 Foreign language versions

Since SUMI started life as a project partly sponsored by the European Commission, it entered a multi-linguistic environment almost from the first stages of its development. With partners in Spain, Italy, the Netherlands and Germany, it was essential to have working versions of the questionnaire in these languages in order for the partners to be able to effectively use the tool with their native user bases.

An Italian version of the earlier CUSI had been prepared and indeed used within an earlier ESPRIT project. However, with SUMI the requirement for translation was not only that the questionnaire items should be faithfully and idiomatically translated, but also that in the target language, respondents' profiles for equivalent systems should be numerically the same. Otherwise we would need to develop a separate standardisation database for each target language.

The process we use is a two-stage one. In the first stage, the questionnaire is translated into the target language by a person with knowledge of human factors and a good knowledge of English. Then the translation is translated back into English by another native speaker of the target language who has not seen the original version. The two source language versions are compared at the HFRG, and where there seems to be a linguistic discrepancy, the original translator is requested for a re-translation. It may be that the back-translation was faulty, as happened on several occasions. In this case, the original translator usually explains the discrepancy, and this is checked separately with the back-translator. The final polish is made by asking speakers fluent in the source language, or native speakers if possible, to check that the translated version does not look like English badly translated. This final step of the first stage should, if possible, include some source language speakers who are familiar with questionnaire work since even in Europe cultural differences exist among peoples' reactions to questionnaires.

The second stage begins with an attempt to locate a sample of 20 or so users who have similar characteristics and who use software similar to what we have on file at the HFRG. A direct statistical comparison, item by item, of these two samples identifies any questions which have been perceived as stronger or weaker in tone than in the source language. If this happens, the item wording can be re-adjusted.

To date the SUMI questionnaire has undergone translation into Italian, Spanish, French, German, Dutch, Greek, Swedish, and USA English. This last is not strictly speaking a translation but an adaptation in order to get rid of Anglo-Irish idioms which had gone un-noticed in the original version. The USA English version was stringently analysed both linguistically and statistically to ensure that meaning change had not taken place on any of the items.

It should almost go without saying, that the business of translation highlights just how volatile a set of statements can be when subjective opinion is involved. Putting the questionnaire up on computer-based media for electronic distribution and devising electronic interfaces for users who fill out the questionnaire does in fact distort the response surface of most of the questionnaire items to a greater or lesser extent. The medium gets in the way. Such a transformation of media should not be undertaken lightly, and where we have done it, we have had to take careful steps to re- standardise the questionnaire by altering item weightings.

3.3 Packaging SUMI

SUMI was packaged with a user handbook, manual scoring stencils and other scoring paraphernalia, and SUMISCO, the scoring program. The current version of SUMISCO works with Microsoft Windows 3.1. SUMISCO carries out all the scoring activities discussed above automatically, and enables export of files which can quickly become evaluation reports to word processors and scored data files to spreadsheets and more sophisticated statistical programs.

The contents of the SUMI box have all been thoroughly tested for quality of use in the HFRG laboratories. The user handbook has been edited carefully in response to comments from the industrial

partners and other data providers during the development phase of SUMI. The form of output from SUMISCO has also been tailored to conform to user expectations.

During the industry-scale validation programme in the last phase of the MUSiC project this care in developing a tool for industrial use and packaging clearly paid off. Partners commented that the SUMI metrics were *the most useful* of all the MUSiC tools (although it must be stressed that SUMI at the time was considerably in advance of the other MUSiC tools in terms readiness for industrial use; this situation has since changed). SUMI also received high ratings from the industrial partners in terms of ease of use for deriving and interpreting the outputs. The validation programme which was carried out by an independent partner within MUSiC is reported in a short project report by Sweeney and Maguire (1994).

4. Validity of SUMI

Three different kinds of validity studies have been conducted with SUMI. Firstly, the industrial partners within the MUSiC consortium used SUMI as part of the industry-scale validation of the MUSiC usability evaluation toolset. A brief account of this activity is given in the MUSiC project final report (Kelly, 1994) and the results are summarised in section 3.3 above.

Secondly, there are now a number of laboratory-based studies which have been carried out in the Human Factors Research Group; and thirdly, studies which have been carried out for industrial clients on a consultancy basis. Laboratory studies to some extent are low in ecological validity; consultancy studies are nearly always commissioned on the understanding of strict confidentiality agreements and are not disclosable in public except in vague outline.

In addition to empirical validation, some theory-based validation has been carried out by comparing the SUMI subscales with the ISO 9241 part 10 dialogue principles. Some considerations arising from this comparison are reported at the end of this section.

4.1 Laboratory studies

4.1.1 Word processors in a work setting

An important early study was carried out by McSweeney (1992) who compared five different word processors in daily use in 11 companies within the Cork region in Ireland. The tasks studied were representative of the uses word processors would be put to in busy offices, and all participants reported having used the word processor under evaluation in their company for at least six months. There were 94 participants in the study who completed the SUMI questionnaire anonymously. Refusal rate was less than 10%.

The word processors studied were: DisplayWrite version 4, Word 4.0 (for the Apple Macintosh), Word 5a (MS-DOS), WordPerfect version 5.1, and a word processor supplied by Wang, which at the time of the study was approximately five years old and had a menu driven interface. Within this sample there were two products which represented relatively old technology, as well as two market leaders and one GUI-based application.

The data was analysed by analysis of variance. The design was a two factor design, 5 (word processor types) x 6 (SUMI Global and five subscales), with repeated measures on the second factor. Since unequal numbers of users evaluated each of the five word processors, an unweighted- means analysis was chosen. The analysis is summarised in table 4.

Table 4: Analysis of variance summary of word processor survey.

| Source | SSQ | df | MS | F |
|---------------------|----------|----|---------|-------|
| A (Word Processors) | 6979.55 | 4 | 1744.89 | 4.55 |
| A Error | 34161.11 | 89 | 383.83 | |
| B (SUMI Scales) | 7393.21 | 5 | 1478.64 | 48.12 |

| | | | | |
|-----------------|----------|-----|-------|------|
| AxB Interaction | 1196.48 | 20 | 59.82 | 1.95 |
| B & AxB Error | 13675.37 | 445 | 30.73 | |
| Total | 66974.48 | 563 | | |

All three F ratios in table 4 are significant at $p < 0.01$. Note especially the significant interaction effect, which indicates that the SUMI scales respond differently to some of the word processors studied, and not simply *en masse*.

A posteriori tests were carried out which yielded the following conclusions.

- For the Global scale, and for Efficiency, Affect and Control, DisplayWrite was rated as significantly less usable than all the other packages; the Wang word processor was also rated as less usable than Word 4. Although not significantly different from Word 5a and WordPerfect, Word 4 was consistently rated as more usable than these two MS-DOS packages.
- For Helpfulness, both versions of Word were rated significantly more usable than the older products. There was no difference between WordPerfect and the older products on this scale.
- For Learnability, Word 4 was rated as significantly more usable than DisplayWrite and WordPerfect; however, the differences between WordPerfect, Word 5a, and the older packages were small and not statistically significant.

These patterns were further amplified by an examination of the ICA results for the five word processors. In summary, SUMI did appear to discriminate between the different word processors and the SUMI subscales picked up trends that had been reported in various independent evaluations of the products studied (for instance, in expert evaluation reports published in PC Magazine). Interestingly, although SUMI showed a difference in favour of the newer packages in the scales of Efficiency, Affect, and Control, the two older packages still maintained a high enough level on Helpfulness and Learnability to suggest that there were reasons other than simple inertia for not replacing them.

4.1.2 Word processors in a laboratory

The Work and Organisational Psychology Unit at the Technical University of Delft carried out a laboratory validation process of the cognitive workload measures devised by that Unit within the MUSiC project. During the process, other MUSiC metrics were also investigated, including the SUMI questionnaire. The work is reported in the document *Measures of Cognitive Workload* by Wiethoff, Arnold and Houwing (1992). This section of the extracts, with their permission, those aspects of their evaluation which pertain to the SUMI questionnaire.

Two word processor packages were assessed: an MS-DOS based product, called M and a Microsoft Windows based product called G. One of the research objectives was to test in a simulated environment tasks and problem states which are known to provoke high workload and stress. From expert assessment it was clear that the differences in the interfaces of these two products would very probably invoke differences in workload, keeping all other factors constant such as task, hardware, environment, etc. On the basis of two independent usability evaluations by experts it was predicted that the G word processor would yield better performance, involve less mental effort, and produce a higher user satisfaction than M for the given tasks.

One evaluation was led by the team from TUD following the seven dialogue principles presented in part 10 of ISO 9241, the other by the HUSAT Research Institute using their own set of usability attributes (such as *ease of use*, *task match*, *learnability/intuitiveness* and so on). Experimental tasks were designed on the basis of the Usability Context Analysis (UCA) method of the National Physical Laboratory; at this stage of the MUSiC project it was called the *Context of Use Questionnaire*. UCA was applied to the job of a university departmental secretary. The experimental task the users had to carry out was to create one document from contributions from several authors, provided to the users either on paper or as a computer file. The output had to conform to certain style specifications. The users were interrupted at three predetermined moments in executing their task by having to answer a telephone. Each user completed the tasks both with the M and G systems, in counterbalanced order. Later analysis showed that order effects were not statistically significant and could safely be ignored.

The users were 26 persons who were working secretaries at the TUD, and they all had a minimum of six months experience with both computers in general and word processors in particular. All users had one hour of guided training with the word processor they were going to use. Other measures taken included psychophysiological effort, subjective measurements of workload and stress, and their sessions were videotaped for further analysis by the NPL performance measurement method, using the DRUM system. The entire evaluation was rigorously controlled by the TUD team, the reader is referred to the document by Wiethoff et al for the details. It represents a paradigmatic case study of applying the entire set of MUSiC user-based metrics and served as a pattern for the two industrial validation studies later in the project.

It should also be noted in passing that the MUSiC method most emphatically does not recommend that the entire set of metrics should be applied in this way for every evaluation. The evaluation at TUD, and at the two industrial partner sites purposely used all the metrics in order to validate and cross-validate the toolset. With regard to the other measures, it was clear that experienced mental effort was lower with the G system than with M. In terms of performance measures, again it was found that G was superior to M, except for more time being spent in the G system on what were considered to be unproductive actions while the users were exploring the functionality of the system. This aspect of performance, called *snag* time was offset by decreased amounts of *search* and *help* time. Devoting more effort in two of the tasks resulted in higher objective mental effort readings for those tasks.

The main effects summary of an analysis of variance conducted on the SUMI data was is given in table 5.

Table 5: Analysis of Variance Summary table for TUD study

| Source | SSQ | df | MS | F |
|---------------|----------|-----|----------|-------|
| A (WP) | 10796.95 | 1 | 10796.95 | 36.72 |
| A Error | 14113.23 | 43 | 294.03 | |
| B (SUMI) | 836.11 | 4 | 209.03 | 6.27 |
| AxB | 311.44 | 4 | 77.86 | 2.34 |
| B & AxB Error | 6396.01 | 192 | | |
| Total | 32488.05 | 249 | | |

In fact, due to an accidental loss of one user's questionnaires, only 25 users were analysed for SUMI (hence total df = 249 in table 5). Note that the data reported by Arnold et al. in their report gives only data from a reduced sample of users whose objective mental effort data were analysed. The main effect of word processor (M compared to G) was significant, as was the difference between the SUMI scales ($p < 0.01$). The interaction of SUMI scales with word processor is just short of statistical significance at $p = 0.05$ and may be overshadowed by the extremely high F ratio for word processors. Note this analysis looked only at the five subscales of SUMI; the difference between the two systems on the Global measure was assessed separately and yielded a student's t statistic value of 5.90 in the expected direction, ($p < 0.01$).

All individual a posteriori comparisons between SUMI sub-scales were statistically significant in the expected directions (i.e., M was poorer on all subscales). The biggest differences were for Learnability and for Affect. Efficiency and Helpfulness showed statistically significant differences which were not as large, and Control yielded the smallest amount of significant difference. From an examination of the means it would seem that the root cause for the large differences was the marked depression of scale scores for the M system. The M system was clearly rated as being poorly suited for the tasks that the users had to carry out in this evaluation.

The ICA analyses supported this verdict, and also supported the main conclusions from the two sets of expert-based evaluation. For M, the organisation of menus and information lists was poor, users felt they had to use too many keystrokes, too many steps were involved for each operation, the system information was not presented in a clear and comprehensible manner. It was not easy to make the software do what was needed, it did not appear to assist the user and in general users felt out of control using the software.

In comparison, with the G system, although users sometimes wondered if they were using the right command selection, and they considered that the initial learning experience was difficult, in general they felt that the system could be mastered and that they could understand and act on the information provided by the software. They reported being able to do non-standard things with ease. The software usually behaved consistently with their expectations and users reported a lack of frustrated feelings with it. With regard to the two sets of expert opinions, the Task Suitability of the G system was considered to be superior. G rated higher than M on Task Match. M had a low level of Self-descriptiveness and Conformity with user expectations. G was rated higher on Learnability and Intuitiveness and on Consistency. Controllability of the M version was considered to be poor and the evaluators noted that the Suitability for Learning of the M version was low. The experts gave the G version higher ratings for Attractiveness. The SUMI results can therefore be seen to accord clearly with the expert evaluations, especially when the ICA is taken into account.

4.1.3 Laboratory-based evaluation of two versions of the same product

Kennedy (1992) reports a study in which two versions of an address-book type database were compared. This is a piece of software that was designed specially for demonstration purposes within the HFRG. It runs on MS-DOS and has a menu and command-line type of interaction style. One version of this system has language and concepts expressed in somewhat old-fashioned computer terms: files have to be explicitly opened and closed, reference is made to records and fields, and direction of search (forwards or backwards) has to be explicitly stated. The system works on the principle that very often, *no news is good news*: if the user is not told of an error, the user must assume that the issued command has taken effect.

The second version of this system has language and concepts expressed in a more user-orientated fashion: a database file is opened automatically, although there is also a possibility to work with another file if the user wants to; reference is made to surnames and information; direction of search works in a forwards direction only, looping back to the start when the end of the database is reached; and the user is notified at all times of the consequences of their actions. The two versions are used normally for didactic purposes and are called loosely the *unfriendly* and *friendly* versions. Other versions of the same software exist, with GUI interfaces of various kinds. These are not evaluated here.

Two different kinds of user were recruited. One group, called the *expert* group, were management science students who had declared experience with information technology products and database management systems in particular. Another group, called the *casual* group had no experience of database management systems and relatively little computer experience but were in the same age cohort as the expert group.

The users were allocated to one of the two versions of the system to form a 2x2 factorial design with independent observations. Each user was given a list of commands and instructions required to use the system, and a demonstration of how each of the commands worked. The users were then asked if they had any further questions, and if they did not, they were given a series of tasks which involved having to use all of the functions of the software. Time taken to complete the tasks was somewhere between four and twelve minutes. After this each participant filled out a SUMI questionnaire. Time and effectiveness data was also taken by Kennedy but will not be reported here.

Kennedy analysed her results using analysis of variance by ranks procedures because of the obvious irregularities in the non-SUMI data, following the principles of Meddis (1984). For each SUMI scale a vector of contrasts is computed and a resultant probability. The probability figure indicates the degree to which the ordering of conditions specified by the vector is attributable to chance, given the obtained average ranks of the samples. Table 6 summarises her results.

Table 6: Lambda vectors for SUMI scales

| | User type: Casual | | Expert | | |
|--------------|-------------------|------------|----------|------------|-------|
| | Friendly | UnFriendly | Friendly | UnFriendly | |
| Probability | | | | | |
| Global | 2 | 3 | 1 | 3 | .0065 |
| Efficiency | 1 | 2 | 1 | 2 | .0051 |
| Affect | 2 | 3 | 1 | 4 | .0056 |
| Helpfulness | 1 | 3 | 2 | 4 | .0067 |
| Control | 3 | 4 | 1 | 2 | .0006 |
| Learnability | 2 | 2 | 1 | 2 | .0020 |

In summary of table 6, for Global, we see significant differences between the friendly and unfriendly versions. The expert users rated the friendly version as being more generally usable than did the casual users.

With regard to subscales, both groups of users rated the friendly version as more Efficient. For Affect, although both groups disliked the unfriendly version, the expert users were more extreme in both their dislike of the unfriendly version and their appreciation of the friendly version. The expert group did not, however, consider that the friendly version was as Helpful as did the casual group, perhaps because the expert group had more background knowledge to fall back on. The expert group felt more in Control in both versions, although the expected differences between versions also showed up.

Casual users were unable to distinguish between levels of Learnability of the two versions, although the expert group considered the friendly version to be easier to learn. Because all users received training beforehand differences in Learnability were slightly surprising. Perhaps the expert group were more able to distinguish between the training they had received and the inherent learnability of the software for carrying out the given tasks. It is stressed that no users evaluated these two systems in a contrastive manner, the design was of the independent subjects type for the user types by friendliness of system interaction.

The Kennedy study shows clearly that SUMI is picking up differences in usability according to interpretable patterns of response. Noteworthy also is the fact that for each system and user group, there were only ten sets of SUMI data, analysed using a statistically conservative ranks-based procedure. This argues well for the precision of the SUMI outputs. No ICA was carried out, but Kennedy remarks that the more experienced end users tended to be more extreme in their reactions, more appreciative of the finer points of the friendly interface and more critical of the usability deficiencies of the unfriendly version.

This last finding was confirmed in an industrial setting on data which we cannot disclose because of confidentiality reasons. The same software was evaluated in two different offices by a different group of users in each office. The users were doing exactly the same kinds of tasks and with more or less the same kinds of training and experience with the product itself, but there were differences in what can only be called depth of experience with other software products between workers in the two offices. Users with greater depth of experience were more critical of poor usability aspects of the software.

4.1.4 Using ICA to get at product improvements

Coleman (1993) carried out a study which was directed specifically at the ICA feature of SUMI. 24 students participated in the study. 12 students who had at least 3 years experience with using a database management system and who were familiar with the underlying theories and concepts constituted the *expert* group. The other 12 students were from the same age cohort but these had never used a database management system except to use the University's library catalogue (DOBIS/LIBIS) and had limited expose to any other kinds of computing. These were called the *novice* group.

The computer system evaluated by both groups of users was the *unfriendly* version of the address-book software studied by Kennedy (1992 -- see section 4.1.3 above). Half of the users filled out a SUMI questionnaire and then took part in a de-briefing based on SUMI ICA; the other half took part in a structured de-briefing exercise without reference to SUMI or the ICA data. Care was taken not to cross-contaminate the two samples.

All the users were requested to say how they thought the software could be improved, by referring as much as possible to specific instances during their experience with the software. Interviews took place at the same table at which the software was running, so users could look at it and use it again if necessary to demonstrate a point. Notes were written down in the users' presence, verbatim. The notes were first sorted into a general list of statements about the software. There were 257 of these from all 24 users. These were then subjected to content analysis. The content analysis scheme yielded 55 categories of statements with an inter-rater reliability of 0.80 or greater. These statements were then independently rated by four final year computer science students (who had no other contact with the project) as being either a *specific design recommendation* or a *vague design recommendation*. There were 48 specific design recommendations according to these raters. Examples of *specific* recommendations are:

- There are no error prevention messages like a noise to warn you that you have done something wrong.
- Change *View/edit/insert* to something like *View the document you have found*.
- Cut out the opening and closing of files (by having one file).

Vague design recommendations were of the sort:

- I couldn't find some of the keys like *delete* and *end* at first.
- It's very hard to say how exactly to improve the software, it's just annoying.
- It's easier to use an ordinary address book.

Coleman now carried out a number of analyses with this data. First, she established that de-briefing users with ICA gave rise to more design recommendations in general (whether *specific* or *vague*). There was no difference between experts and novices on the number of design recommendations put forward within the SUMI-ICA condition, or the Interview condition. This finding simply demonstrated the superior effects of cued recall. However, taking only the 48 specific design recommendations, Coleman found that the probabilities of a design recommendation that could be classified as specific emerging from experts, novices, SUMI and interviews showed some interesting differences. These probabilities are shown in table 7.

Table 7: Probability of eliciting specific design recommendations

| | Expert | Novice | Combined |
|-----------|--------|--------|----------|
| SUMI | 0.447 | 0.342 | 0.790 |
| Interview | 0.166 | 0.044 | 0.210 |
| Combined | 0.613 | 0.387 | 1.000 |

Thus SUMI greatly improves novices' ability to offer specific design recommendations, and it improves experts' ability to a lesser extent. No assessment was made of the severity of the *errors* that lay behind the design recommendations, and we should note that in practice, some sort of guidelines or memory prompts would be used to assist evaluations, both for novices and for experts. However, it is not clear that such guidelines would be equally comprehensible for experts and for novices.

4.2 Consultancy-based studies

With regard to industrial consultancies, three case study scenarios which have been edited and idealised to protect the companies involved are briefly sketched to indicate the range of uses to which SUMI has been put. Company X used SUMI to do a company-wide evaluation of all the office systems software they were using. The information from the usability profiles assisted the company in identifying software that needed replacing, and to draw up a staff re-training strategy. The company specifically had in mind the European Directive on Health and Safety for Work with Display Screen Equipment (EEC, 1990)

when they undertook this task. SUMI enabled them to fulfil about a third of the requirements mentioned in the directive.

Company Y was about to purchase a large data entry system. They had narrowed the field down to two possibilities. The company wished to let the actual end users experience both systems to assist in the decision making process, since this was an area in which on-site data entry was only starting to be introduced. The company, in participation with the two potential suppliers, arranged for a selection of staff to visit sites where the systems had been installed on two successive weekends. Both systems were evaluated using SUMI. The users involved felt that SUMI had sharpened their perceptions of what they should be looking for in a computer system. Although the eventual system purchased was the one with the lower usability ratings, the company used the SUMI profiles (which were exceptionally low for Learnability on the chosen system) to successfully support their bid for a large training and support element in the package.

Company Z who was a software vendor had developed a new GUI version of their successful software package in response to a number of comments on the usability of the interface of the previous version. The package was a client-server interface to a large database, and the end users of the package were using it to respond to real-time queries from the public. When the two versions of the interface were evaluated, it turned out in fact that in many of the SUMI scales, the new version was worse than the old. After checking for familiarity effects to no avail, the evaluators found that the real problem lay in the fact that the interface had become too complicated in the new version, and took too long to operate in answer to a query. The new version was not released, and instead, was scheduled for a further process of re-design to take into account what had been learned about it during the evaluation. Had the product been released without evaluation, it would have resulted in a costly loss of confidence in the company.

4.3 Comparison with ISO 9241 part 10

When we compare the SUMI sub-scales with the seven dialogue principles of ISO 9241 part 10 we see immediately that four of the SUMI subscales have an obvious correspondence. This enables the analyst to measure at least four of these dialogue principles directly and empirically with reference to the opinions of an end-user sample.

SUMI Helpfulness and the ISO principle of Self-Descriptiveness are clearly related when one compares the descriptions of both. SUMI Control and ISO Controllability are also clearly related, as are SUMI Learnability and the ISO principle of Suitability for Learning. SUMI Efficiency is also strongly related to the ISO principle of Suitability for the Task.

Less directly, perhaps, the SUMI scale of Affect is related to the ISO principle of Conformity with User Expectations, when the wordings of the items making up the Affect scale are laid side-by-side with the description of the corresponding ISO principle, although the ISO construct may also entail a component of SUMI's Efficiency. There remain the ISO principles of Error Tolerance and Suitability for Individualisation which may not be directly related to any one specific SUMI subscale. Although the SUMI Helpfulness scale may be considered to relate to aspects like error recovery, the wording of the actual items in the Helpfulness scale would suggest that end users are concerned with helpfulness issues beyond those specifically related to errors. SUMI does not have a scale which could directly relate to the ISO principle of Suitability for Individualisation: in fact, some items referring to this concept were eliminated early on during the development of SUMI because they did not appear to form a coherent factor on their own, nor did they load strongly on any of the five emergent factors that subsequently became the SUMI sub- scales.

A recently published study by Beimeel et al. (1994) gives another perspective on the above comparison. This study analysed the opinions of 90 Human Factors experts in nine countries about various aspects of the ISO 9241 standard. It is an ambitious undertaking, and the report deserves to be carefully studied by anyone wishing to evaluate the quality of use of software. A small part of this study's results concerns the question to what extent did the experts consider each of the seven principles *essential* (versus *superfluous*). Suitability for the Task, Self-Descriptiveness, Controllability, Conformity with User

Expectations, and Error Tolerance were evaluated as essential by the majority of the Human Factors experts. Suitability for Learning was rated as a little less essential, but Suitability for Individualisation was rated least essential by a good margin.

The congruence of the empirical evidence derived from end users' constructs (SUMI) and the commentary by Beimel and his associates summarised above provides strong reasons for claiming high construct validity for the existing SUMI sub-scales as well as for the ISO part 10 principles (with the exception perhaps of Suitability for Individualisation on the part of the ISO standard.)

5. The SUMI Questionnaire as at Present

SUMI is applicable to any software system which has a display, a keyboard or other data entry device, and a peripheral memory device such as a disk drive. It has also been used successfully for evaluating the client side of client-server applications. SUMI is indicated by Preece et al. (1994) as a standard method for assessing user attitudes, and by Dzida et al. (1993) as a way of achieving measurement of user acceptance in the context of the Council Directive on Minimum Safety and Health Requirements for Work with Display Screen Equipment (EEC, 1990). Davies and Brailsford (1994) also recommend the use of SUMI in their series of guidelines for multi-media courseware developers. The HFRG have received notice by other groups of authors that they intend to recommend SUMI in publications relating to the evaluation and development of information technology and indeed the latest draft of the ISO 9241 Part 11 references SUMI.

The minimum user sample size needed for an analysis with tolerable precision using SUMI is on the order of 10 - 12 users, although evaluations have been carried out successfully with smaller sample sizes. However, the generalisability of the SUMI results depends not so much on the sample size itself, but the care with which the context of use of the software has been studied and the design plan has been made. As summarised by Macleod (see elsewhere in this volume) this involves identifying the typical users of the software, the goals which they typically wish to achieve, and the technical, physical and organisational environments in which the work is carried out (for prototype systems, this involves determining the future context of use). The design plan requires an adequate sampling of the context of use.

With regard to the requirement that there be a working version of the product, this does not turn out to be such a serious limitation after all. Most software is created on the basis of improvements or upgrades to a previous version, or in response to a market opportunity created by gaps in competitive products. Usability evaluation can therefore feed into the earliest stages of system specification, as well as enabling the setting of usability targets to be achieved by the new system. Many companies now use some kind of rapid prototyping strategy, especially with GUI environments, and the SUMI questionnaire lends itself ideally to this kind of development work as it is short (it takes 5 minutes to complete, at maximum) and does not require a large user sample. Redmond-Pyle and Moore recommend incorporating SUMI into a comprehensive GUI development methodology (Redmond-Pyle and Moore, in press.)

The questionnaire is administered in paper-and-ink format. Some computerised versions exist, but it is important to re-standardise the questionnaire for each implementation, as differences between implementations can appear in the response profiles of users. Besides which, most industrial users at present at any rate, prefer to use the paper-and-ink format. The package comes with 50 questionnaires supplied in the language of the evaluator's choice. The standard default is UK English, but the questionnaire is also available in US English, French, Italian, Spanish, German, Dutch, Greek and Swedish. Evaluators may order more packages of questionnaires, or additional packs if they wish to carry out multi-lingual comparisons.

SUMI may be administered and scored by experienced psychometricians, who will usually need training only in the interpretative aspects of the analysis. SUMI administrators with a psychological background will in addition need training in scoring procedures and interpretation, with usually a review of the statistical procedures involved. Other HCI personnel who wish to use SUMI are strongly advised to also take a refresher course, if they have not already covered such issues recently, in evaluation study design.

With appropriate training SUMI has been used successfully by software engineers in an industrial setting as well as by more psychometrically aware researchers in academia. In addition to the SUMI package training and certification services are also available, as well as evaluation consultancy based on the SUMI questionnaire in conjunction with other usability tools from the MUSiC toolset when appropriate. Human Factors and Software Quality consultants in many countries have taken up SUMI as part of their professional activities with enthusiasm.

6. References

- Anastasi, A, 1968 *Psychological Testing* (3rd Ed). Collier-Macmillan Ltd, London.
- Bailey J and Pearson S, 1983, Development of a tool for measuring and analysing computer user satisfaction. *Manag Sci*, 29, 530-545.
- Baroudi JJ, and Orlikowski WJ, 1988, A Short Form measure of User Information Satisfaction: A Psychometric Evaluation and Notes on Use. *J Manag Info Syst*, 4.4, 45-59.
- Beimel, J, Schindler, R, and Wandke, H, 1994, Do human factors experts accept the ISO 9241 part 10 -- Dialogue Principle -- standard? *Beh and Info Technol*, 13.4, 299-308.
- Coleman, N, 1993, SUMI (Software Usability Measurement Inventory) as a knowledge elicitation tool for improving usability. Unpublished BA Honours thesis, Dept. Applied Psychology, University College Cork, Ireland.
- Chignell, M, 1990, A taxonomy of user interface terminology. *SIGCHI Bull*, 21.4, 27-34.
- Chin, JP, Diehl, VA, and Norman, KL, 1988, Development of an instrument measuring user satisfaction of the human computer interface. *Proc CHI '88*, 213- 218, ACM.
- Davies P and Brailsford T, 1994, *New Frontiers of Learning: Guidelines for Multi-Media Courseware Developers*, vol. 1, Delivery, Production and Provision. Dept. Life Science, University of Nottingham, UK.
- Deese, D, 1979, Experiences measure user satisfaction. *Proc Comp Meas Group ACM Dallas*, Dec 1979
- Doll, WJ, and Torkzadeh, G, 1988, The measurement of end-user computing satisfaction. *MIS Quart*, 12, 259-274.
- Dolotta, TA, Bernstein, RS, Dickson, S (Jr), France, NA, Rosenblatt, BA, Smith, DM, and Steel, TB (Jr), 1976, *Data Processing in 1980-1985, A Study of Potential Limitations to Progress*. Wiley-Interscience, NY.
- Dzida, W., Herda, S. and Itzfeldt, WD, 1978, User Perceived Quality of Interactive Systems, *IEEE Trans Softw Eng* SE-4.4 270-276.
- Dzida, W., Wiethoff, M, and Arnold, AA, 1993, *ERGOGuide: the Quality Assurance Guide to Ergonomic Software*. Delft University of Technology, Dept. of Work and Organisational Psychology, PO Box 5050, 2600 GB Delft, the Netherlands.
- EEC, 1990, Minimum Safety and Health Requirements for Work with Display Screen Equipment (90/270/EEC) *Official Journal of the European Communities* No. L 156, 21/6/90.
- Flanagan, J, *The critical incident technique*. In: Cook, M, *Personnel Selection and Productivity*, Wiley & Sons, Chichester.
- Grudin, J, 1991, Systematic sources of suboptimal interface design in large product development organisations. *Hum Comput Interact* 6.2, 147-196.
- Grudin, J, 1992 Utility and usability: research issues and development contexts. *Interacting with Comput*, 4.2, 209-217.
- Holcomb, R, and Tharp, AL, 1991, Users, a software usability model and product evaluation. *Interacting with Comput*, 3.2, 155-166.
- Humphreys, M, Pike, R, Bain, J, and Tehan, G, 1988, Using multilist designs to test for contextual reinstatement effects in recognition. *Bull Psychonom Soc*, 26.3, 200-202.
- Igbaria, M., and Parasuraman, S., 1991, Attitudes towards Microcomputers: Development and Construct Validation of a Measure. *Int J Man Mach Stud.*, 35.4, 553-573.
- Igbaria, M., and Nachman, S.A., 1991, Correlates of User Satisfaction with End User Computing: an Exploratory Study. *Info and Manag*, 19, 73-82.
- ISO 9241 (Ed), 1991, ISO 9241 Ergonomic Requirements for Office Work with Visual Display Terminals, Part 10, Dialogue Principles (CD).

- Ives, B.S., Olson, M., and Baroudi, J., 1983, The Measurement of User Information Satisfaction. *Comm ACM*, 26, 530-545.
 - Kelly, M. (Ed), 1994, *MUSiC Final Report Parts 1 and 2: the MUSiC Project*, BRAMEUR Ltd, Hampshire, UK.
 - Kirakowski, J., 1987, The Computer User Satisfaction Inventory. IEE Colloquium on Evaluation Techniques for Interactive System Design, II, London.
 - Kirakowski, J., and Corbett, M., 1988, Measuring User Satisfaction, in Jones, D.M., and Winder, R., *People and Computers*, vol. IV, Cambridge University Press, UK.
 - Kirakowski, J., and Corbett, M., 1990, *Effective Methodology for the Study of HCI*, North-Holland, Amsterdam.
 - Lewis, J.R., 1991, Psychometric Evaluation of an After-Scenario Questionnaire for Computer Usability Studies: the ASQ. *SIGCHI Bull*, 23.1, 78-81.
 - Lucey, N.M., 1991, More than Meets the I: User-Satisfaction of Computer Systems. Unpublished thesis for Dipl. Applied Psychol, University College Cork, Ireland.
 - Meddis, R., 1984, *Statistics using Ranks: a Unified Approach*. Blackwell, UK.
 - McSweeney, R., 1992, SUMI -- A psychometric approach to software evaluation. Unpublished MA (Qual) thesis in Applied Psychology, University College Cork, Ireland.
 - Molich, R., and Nielsen, J., 1990, Improving a human-computer dialogue. *Comm ACM*, 33.3, 338-344.
 - Nielsen, J., 1992, *Finding usability problems through heuristic evaluation*. In: Bauersfeld, P, Bennet, J, and Lynch, G (Eds.): *CHI'92 Conference Proceedings*, ACM Press, NY, 373-380.
 - Opperman, R, Murchner, B, Paetau, M, Pieper, M, Simm, H, and Stellmacher, 1989, *Evaluation von Dialogsystemen, Der Software-ergonomische Leitfaden EVADIS*, Walter de Gruyter, Leiden.
 - Opperman R, Murchner, B, Reiterer, H, and Kock, M, 1992, *Software-ergonomische Evaluation, Der Leitfaden EVADIS II*, Walter de Gruyter, Leiden.
 - Preece, J, Rogers, Y, Sharp, H, Benuyon, D, Holland, S, and Carey, T, 1994, *Human Computer Interaction*. Addison-Wesley.
 - Ramamurthy, K., King, W.R., and Premkumar, G., 1992, User Characteristics-DSS Effectiveness Linkage: an Empirical Assessment. *Int J Man Mach Stud*, 36, 469-505.
 - Ravden, SJ and Johnson, GI, 1989, *Evaluating usability of Human Computer Interfaces: a practical method*. Ellis Horwood, Chichester.
 - Redmond-Pyle, D., and Moore, A., 1995 (in press), *Graphical User Interface Design and Evaluation: a Practical Process*. Prentice Hall International.
 - Reiterer, H, and Oppermann, R, 1993, Evaluation of user interfaces: EVADIS II -- a comprehensive evaluation approach. *Behaviour and Information Technology*, 12.3, 137-148.
 - Saunders, CS and Jones, JW, 1992, Measuring performance of the information systems function. *J Manag Info Syst* 8.4 63-82.
 - Shneiderman, B., 1987, *Designing the User Interface: Strategies for Effective Human-Computer Interaction*, Addison-Wesley, Reading, MA.
 - Shneiderman, B., 1992, *Designing the User Interface: Strategies for Effective Human-Computer Interaction*, 2nd ed., Addison-Wesley, Reading, Mass.
 - Spenkelink, GPJ, Beuijen, K, and Brok, J, 1993, An instrument for the measurement of the visual quality of displays. *Behaviour and Information Technology*, 12.4 249-260.
 - Sweeney, M, and Maguire, M, 1994 Industry-Scale Validation Programme, MUSiC ESPRIT Project 5429 document code HUSAT/DV3.
 - Thimbleby, H, 1990, You're right about the cure: don't do that. *Interacting with Comput*, 2.1, 8- 25.
 - Wiethoff, M, Arnold, A, and Houwing, E, 1992, Measures of Cognitive Workload. MUSiC ESPRIT Project 5429 document code TUD/M3/TD/2.
 - Wong, G.K., Rengger, R., 1990, The Validity of Questionnaires designed to Measure User-Satisfaction of Computer Systems. National Physical Laboratory report DITC 169/90, Teddington, Middx., UK.
-