User and usability testing - how it should be undertaken?

Merle Conyer
Digital Equipment Corporation
merle_conyer@yes.optus.com.au

Usability evaluation is the analysis of the design of a product or system in order to evaluate the match between users and a product or system within a particular context. Usability evaluation is a dynamic process throughout the life cycle of a product or system. Conducting evaluation both with and without end-users significantly improves the chances of success. Six usability evaluation methods and six data collection techniques are discussed, including advantages and limitations of each. Recommendations are made regarding the selection of particular evaluation methods and recording techniques to evaluate different elements of usability.

Usability refers to the extent to which a product[1] can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use and at an acceptable cost (Bevan; 1995a, p.885; Mack and Nielsen, 1994, p.3; Sweeney, Maguire and Shackel: in Stanney and Mollaghasemi, 1995, p.387). The purpose of evaluating for usability is to find usability problems and then make recommendations to fix these problems, and so improve the usability of the design.

Two main classes of usability evaluation methods can be differentiated (Ziegler and Burmester: in Anzai et al, 1995, p.899). One class focuses on users of a particular product and aims to determine usability by assessing users using a product. This approach will be referred to as *user testing*. The other methods are designed to support human factors engineers with evaluating the usability of a product and will be referred to as *usability testing*.

User and usability testing

No matter how much analysis has been done in designing an interface, experience has shown that there will be problems that only appear when the design is tested with users (Lewis and Rieman, 1994). The user's experience of a product's usability is the ultimate test of quality (Whiteside et al., 1988. p792).

User testing is based on the analysis of user behaviour during the use of the product to be evaluated. Therefore user testing requires an understanding of the actual user profiles, their tasks and the contexts in which the tasks are performed. As one important objective of user testing is to ensure that user differences are accommodated in order to minimise the variation in user performance, it is therefore important that user testing is done with a sample of people whose background knowledge and expectations approximate those of the real users. During user testing, users should be allowed to work in realistic conditions, without interruption from an observer, in order to accurately replicate the intended context of use (Bevan, 1995b, p.354).

There are a number of reasons to consider evaluating usability without users. Users' time is almost never unlimited or a free resource. Users can find it difficult to visualise how a product could behave differently and they therefore tend to evaluate according to what already exists, rather than to what is possible. Some usability criteria will only be reliably identified or articulated by trained human factors engineers. Further, some methods of usability evaluation require skill and training in their use.

At least three evaluators with a mix of experience and expertise are required for user and usability testing because fewer will not identify all the usability problems (Nielsen: in Nielsen and Mack, 1994, p.33).

Embedding usability evaluation in all phases of the life cycle

Engineering for usability requires early specification of usability goals. Usability is built into a new product from the analysis phase of a project by identifying and analysing the critical features of and interactions between users, their tasks and the product. Specific contexts in which usability is to be measured should also be identified. These usability goals can then be used to interpret the findings from the user analysis and to identify the goals and constraints that will direct the design and set criteria against which a design can be tested once it is built.

As part of the design phase, it is recommended that prototypes be developed and tested, iterating on the design as a result of the outputs from usability testing. It is important to begin usability evaluation at the earliest phases of design because, if left until just before release, there will be little chance to make any significant design changes.

Testing formally for compliance with usability specifications takes place in the testing phase.

Another level of formative evaluation involves consideration of user acceptance In user acceptance testing it is recommend that users test not just the product but all parts of the package that the users will receive, such as training, written procedures, forms, manuals, computer-based

training and on-line help (McManus and Hammond, 1991, p.101). This integrated approach ensures that there is no mismatch between the different components and highlights the users' perspective of the whole product rather than a number of the parts. For this testing, a prediction is needed of the organisational and task changes that will occur as a result of the introduction of the new product.

Once the new product has been implemented, it is useful to follow up with contextual summative evaluation in order to understand the actual learning process, usability issues and use of the product by novices and experts in a realistic work context.

There are a number of benefits for usability evaluation if it is iterative and considered in all phases of the development life cycle. Iterative design helps with the management of product development and so reduces the risk of projects going off track. Early testing can detect unclear or unreasonable usability goals. Usability objectives can help to facilitate communication and decision-making between human factors engineers and product designers. Also, usability testing allows developers to obtain and appreciate a user perspective of their product.

Usability evaluation methods

There are a variety of usability evaluation techniques available which serve different purposes and which involve a combination of user and usability testing. In this paper the following methods will be discussed:

- Heuristic evaluation
- Pluralistic walkthroughs
- Formal usability inspections
- Empirical methods
- Cognitive walkthroughs
- Formal design analysis.

Heuristic evaluation
The Heuristic evaluation method uses a predefined list of recognised heuristics (usability principles) to identify usability problems so that they can be attended to in an iterative design process.

Method: Human factors engineers and/or end-users independently examine the interface and judge its compliance with a predetermined set of heuristics. It is recommended that each evaluator work through the interface at least twice, the first time to get a feel for the flow of the interaction and the second time to focus on the specific interface elements within the context of the larger whole (Nielsen: in Mack and Nielsen, 1994, p.29). Observers can offer help to evaluators when they are clearly having difficulty and after they have commented on the usability problem they are experiencing. The evaluators' comments can be recorded either by

themselves or by an observer. The human factors engineer has the responsibility of interpreting their own comments; the observer has the responsibility of interpreting the users' actions and comments. These results are then aggregated.

A debriefing session is then held with all evaluators, observers and representatives of the design team to brainstorm possible ideas to address the major usability problems, as well as to discuss the positive elements of the interface design. A prioritised list is then drawn up of all usability problems with reference to the heuristics that were not followed in the design, and with a time and cost estimate to correct each problem. Priority is determined according to the frequency and impact of the problem, and if the problem can be overcome in another way, eg. with training.

Advantages

- The method is quick to learn and easy to use.
- Sessions are recommended to be no longer than two hours, so it is therefore not time-consuming for evaluators.
- Results are available quickly.
- The method is inexpensive to implement.
- Evaluators represent different knowledge domains.
- Independent evaluation means that findings are unbiased.
- The method can be used to identify problems early in the design process.
- Evaluation can be performed with specifications, paper prototypes, simulations or full functioning prototypes.
- Participatory design leads to collaboration to meet the criteria of all parties involved in the evaluation.
- Developers will be better prepared to create more usable interfaces in the future.

Limitations

- If a debriefing session is not held, usability problems are identified with no indication of how they can be fixed.
- The evaluation is not conducted in the context in which the task would normally be performed.
- The method can preclude evaluators from browsing and exploring which can often lead to additional learning about the usability of the interface.
- The evaluation is narrowly focused and therefore identifies more specific problems and not general problems.

Pluralistic walkthroughs

The goal of this method is to systematically review the usability of an interface and its flow from a task-based user-centred perspective whilst at the same time considering the design constraints.

Method: In the context of task-based scenarios, end-users, product developers and human factors engineers evaluate a product from the perspective of the end-user. The evaluators write down sequentially each action they would take when pursuing a designated task. A group

discussion then follows, with end-users presenting their information first. Subject matter experts are available at all stages for domain-specific questions.

Advantages

- The method is quick to learn and easy to use.
- The method can be used to identify problems early in the design process.
- Evaluation can be performed with paper prototypes, simulations or full functioning prototypes.
- Evaluators represent different knowledge domains.
- Independent evaluation means that findings are unbiased.
- Allows for rapid test-redesign-test usability engineering.
- As an outcome of this method of evaluation, subject matter experts can identify requirements for supporting documentation or help.
- Participatory design leads to collaboration to meet the criteria of all parties involved in the walkthrough.
- Developers will be better prepared to create more usable interfaces in the future.

The method is inexpensive to implement.

Limitations

- The evaluation is not conducted in the context in which the task would normally be performed.
- The process must progress as slowly as the slowest reviewer - this can mean that the participants do not get a good grasp of the flow of the interface (Bias: in Mack and Nielsen, 1994, p.69).
- The method can preclude evaluators from browsing and exploring which can often lead to additional learning about the usability of the interface.
- Because the walkthrough is task-based it is narrowly focused and therefore identifies more specific problems and not general problems.

Formal usability inspection

Usability issues are reviewed within the context of specific user profiles and defined goal-oriented scenarios by applying a task performance model and heuristics.

Method: This method captures how evaluators *perceive* the information, *plan* to use the information, *decide* how to proceed and *perform* the selected action. A six-step process is used, namely *Planning; Kick-off meeting*, when the team comes together for the first time; *Preparation*, when the evaluators review independently; *Review*, to discuss the aggregated usability issues; *Rework*, when solutions are found and implemented; and *Follow-up*, to determine the effectiveness of the evaluation process.

There are clearly defined participant responsibilities, namely: *Moderator*, who manages the process; *Design owner*, who is responsible for representing and then upgrading the product being inspected; *Evaluators*, who find and report usability problems (such as designers, documentation

specialists and human factors engineers); and *Scribe*, who records all identified problems and decisions.

Advantages

- Independent evaluation means that findings are unbiased.
- Evaluators represent different knowledge domains.
- The outcome is a list of usability problems and implemented solutions.
- There will be fewer problems encountered by users when user testing is conducted.
- The method evaluates both cognitive processing and behavioural tasks.
- Participatory design leads to collaboration to meet the criteria of all parties involved in the inspection.
- Developers will be better prepared to create more usable interfaces in the future.
- Efficiently managed process.

Limitations

- End-users are generally not involved in this type of evaluation.
- Evaluation is not conducted in the context in which the task would normally be performed.
- The method can preclude evaluators from browsing and exploring which can often lead to additional learning about the usability of the interface.
- This method is less effective when used in the earlier stages of product development.

Empirical method

Data is collected in an experimental test to prove or disprove an hypothesis; for example, the number of correct responses and errors made by a user under controlled conditions.

Method: An hypothesis is posed based on a set of objective measures for the evaluation. A plan for how the measures are to be collected is then determined. The next step is to find subjects for the test, to collect the data and to analyse the data to determine if the proposed hypothesis has been proven.

Advantages

- Effective for establishing cause and effect.
- Effective for addressing a specific question or problem through focused testing.

Limitations

- It can be time consuming and expensive to conduct good experimental work (Karat, 1988, p.899).
- Human action derives its meaning from the context in which it occurs (Whiteside et al., 1988, p.806). Therefore if empirical usability evaluation takes place, for example in a laboratory and not in the anticipated context of use, it is then important to try and understand how altering the context may alter the meaning and significance of the observed behaviour.

- Requires a skilled practitioner trained in empirical methods.
- Can be very time-consuming as empirical evaluation sessions may need to be repeated numerous times in order to validate a hypothesis and to remove any doubt of experimenter bias.
- This method cannot be used until, at a minimum, a robust prototype is available. Generally does not involve the product designers, therefore can be more difficult to influence the design.
- The method can preclude evaluators from browsing and exploring which can often lead to additional learning about the usability of the interface.
- A problem in software evaluation is that the nature of the interface changes so rapidly (thus changing the hypothesis) that the effort and expense associated with such formal testing is rarely justified (Karat, 1988, p.899).

Cognitive walkthroughs

Cognitive walkthroughs are used to evaluate the ease of learning to use a product, particularly by exploration (Wharton et. al.: in Mack and Nielsen, 1994, p.l08). The method is a formalised way of imagining people's thoughts and actions when they use a product interface for the first time (Lewis and Rieman, 1994, Chapter 4.1).

Method: Cognitive walkthroughs focus most clearly on problems that users will have when they first use an interface, without training. The method uses an explicitly detailed procedure to simulate a user's problem-solving process at each step, checking to see if the user's goals and memory for actions can be assumed to lead to the next correct action (Mack and Nielsen, 1994, p.6). There are three phases in the procedure, namely *Preparatory*, when the analysts agree on the input conditions for the walkthrough, such as type of users, tasks and action sequence for each task; *Walkthroughs*, which can be an individual or group process; and *Analysis.*

Advantages

- Effective method for predicting problems that affect users' performance.
- Evaluation can be performed with paper prototypes, simulations or full functioning prototypes.
- Designers are forced to consider the user's background knowledge, the user's goal structure and the cognitive complexity required for a user to use the product.
- Captures cognitive processes of both novice and expert users.
- Independent evaluation means that findings are unbiased.

Limitations

- The method continually interferes with the interaction.
- People not trained in cognitive psychology may find it difficult to decompose tasks into a collection cf sub-tasks.
- Narrow focus as focuses primarily on one attribute of usability, namely ease of learning.
- Can be a tedious and time-consuming process.
- One cannot get comparable measures of task time.

Formal design analysis

Formal design analysis techniques aim at improving the design process. Examples are the 'Goals, Operators, Methods and Selection rules' (GOMS) model developed by Card et al. (1983: in Eberts, 1994) and the 'Natural GOMS Language' (NGOMSL) model developed by Kieras (1988: in Eberts, 1994).

Method: Formal design analysis is based on the premise that understanding of the requirements of the task to be performed is the key to understanding behaviour. Tasks to be performed by an expert user are decomposed into *goals* (a series of cognitive and motor components), *operators* (actions that a user executes), *methods* (sequences of steps) and *selection rules* (needed if more than one method is available to accomplish a goal). Algorithms are then applied and each design is rated with a single number. Alternative design possibilities are then compared based on the numerical result.

Advantages

- These techniques provide assistance that will enable accurate design decisions to be made without necessitating costly user testing (Karat, 1988, p.900).
- Adequate for analysing well learned domains in which there is a minimum of problem- solving behaviour required of the user.
- The method can be used to identify problems early in the design process.
- Inexpensive as can be performed by a single person.
- Useful for comparing the usability of different designs.

Limitations

- Assumes that all cognitive operations are of equal difficulty.
- No decision-making or problem-solving process constraints are taken into account.
- The method misses many key components of behaviour that must be considered in interface design such as learning the task, error behaviour and transfer of learning to other products.
- Only suitable for analysing expert behaviour.
- Complex to learn.

Recording methods

When using the above usability evaluation methods, there are a variety of recording methods that can be used to capture data. In this paper the following data collection methods will be discussed:

- Verbal reports
- Concurrent think-aloud method
- Questionnaire
- Video analysis
- Auto-logging programs and audit trails
- Software support

Verbal reports

Users provide a verbal report soon after completing their evaluation. This information can then be informally reviewed or formally classified into categories for evaluation.

Advantages

- An excellent way to gain access to the contents of someone's short-term memory (Karat, 1988, p.897).

Limitations

- Requires extensive analysis. Information is degraded if it is not collected close to the time that the activity is performed.
- A significant number of subjects will not provide useful verbal reports (Karat, 1988, p.898).
- Interventions by the evaluator can reduce the accuracy of the report.

Concurrent think-aloud method

Evaluators verbalise their thoughts while interacting with a product. The purpose of this method is "to show what the users are doing and why they are doing it *while* they are doing it, in order to avoid later rationalisations" (Nielsen: in Vora and Helander, 1995, p.375).

Advantages

- Particularly useful for identifying usability problems with decision support tools because emphasis is on reasoning rather than on routine performance.
- Does not rely on memory recall.

Limitations

- People are not used to thinking aloud so the method can be intrusive and distracting.
- As thinking aloud is not something that people are used to doing, subjects rarely give quality think-aloud reports without prompting (Karat, 1988, p.897).
- Requires extensive analysis.
- The analysis of the information can be difficult.

Questionnaire

Questionnaires can be composed of items that address information and attitudes. It is important to keep questions specific rather than general and to ask questions about actual product experience rather than hypothetical questions about possible product changes (Root and Draper, 1983; in Karat, 198S, p.896).

Advantages

- Easy and inexpensive method for obtaining and analysing data.
- Scaled questions are easy to quantify.

Limitations

- Scaled questions may be easier to quantify but they limit information that may be obtained through open-ended questions.
- Questionnaires are usually administered after using the product, rather than while the evaluator is in the process of using the product, thus some information may be lost.

Video analysis

One or more videos can be used to capture data about user interactions. For example, in a software usability evaluation, three different video cameras could be used to capture the user's keyboard actions, the screen activity and the user's verbal and non-verbal responses. Video Analysis is then used as a tool in the process of interpreting what usability problems occur and why. Even more powerful is to use the video to create a multimedia document which includes annotations of the usability problems.

Advantages

- Useful for demonstrating usability problems to designers who find it impossible to believe that others cannot use their product has more impact and therefore more influence than charts or statistical summaries of usability problems (Chignell et al, 1995, p.325).
- Allows designers to review usability problems without having to leave their offices.

Limitations

- While large amounts of video data can be collected, relatively little of this video is actually analysed as the extra time spent to analyse videos is often judged to be spent more effectively in carrying out further usability tests to identify more usability problems (Chignell et al, 1995, p.325).

Auto-logging programs and audit trails

Auto-logging programs can be used to track user actions with respect to duration and frequency of use, like number of keystrokes, button clicks, requests for help, duration and path of errors.

Advantages

- Cost effective way to gather information regarding user actions.
- The evaluator is freed from routine work and can concentrate on the evaluation.

Limitations

- Statistically analysing such data can be very complex.
- High level behaviours and trends must be deduced from low level artefacts such as key strokes (Yoder, McCracken and Akscyn: in Wilson and Rosenberg, 1988, p.871).
- Frequently it is necessary to combine this type of evaluation with video analysis to reassemble a complete picture of what the user was doing.

Software support
Software can be designed to support the evaluator during the evaluation process and to provide an assessment summary. For example, during the evaluation of a system user interface, the test items are presented on the screen with accompanying usability criteria and a rating scale. The evaluator selects the usability criteria, giving each a rating and writes an explanation of each rating. The software calculates an average mark for each criterion and sorts the results by usability components (Reiterer, H. and Opperman, R., 1995, pp 364-366).

Advantages

- The evaluation criteria as well as the reasoning behind them are captured.
- The context of use can be included in the evaluation.
- Reduces opportunities for omission of data.

Limitations

- The method continually interferes with the interaction.
- Statistically analysing the explanations can be very complex.

Considerations when selecting an evaluation methodology

Different usability evaluation methods address different usability problems. The following should be considered when selecting a usability evaluation methodology:

- The objectives of the usability evaluation.
- The nature of the criteria for evaluation.
- How open-ended or constrained the evaluation should be; eg, should the interface be evaluated according to predefined scenario-based end-user tasks, or should it be more open-ended so that the users generate their own goals and paths.
- Which method to use, taking into account the goals of the method, the appropriateness of the method, and how easy the method is to use and learn.
- Which methods to use in combination.
- Usability and learnability of the methods.
- The number of evaluators required to provide a reliable result.
- Evaluator expertise and experience.
- The availability of end-users.
- Benchmarks - what to compare the evaluation results against.
- The development constraints of the project and potential trade-offs.
- How easily and efficiently the results can be communicated to the designers and product owner.
- Available time.
- Available budget.
- Cost benefit of the usability test.

Which method to choose?

The table that follows suggests different methods and data collection tools that can be considered for different evaluation purposes.

If the purpose of the usability evaluation is to evaluate...	then consider the... methodology	using the... recording method.
the ability of the user to carry out a task using a product in a particular context	Formal Usability Inspection	Verbal Reports Concurrent Think-Aloud Video Analysis Software Support
how easily users can carry out a task	Pluralistic Walkthrough Formal Usability Inspection Cognitive Walkthrough Formal Design Analysis	Verbal Reports Concurrent Think-Aloud Questionnaire Video Analysis Auto-Logging Programs and Audit Trails
how quickly users can carry out a task	Empirical Formal Design Analysis	Video Analysis Auto-Logging Programs and Audit Trails
the overall quality and acceptance of a product	Heuristic Evaluation	Verbal Reports Questionnaire Software Support
problems with using a product	Pluralistic Walkthrough Formal Usability Testing Cognitive Walkthrough	Verbal Reports Concurrent Think-Aloud Video Analysis Questionnaire Auto-Logging Programs and Audit Trails Software Support
how easy it is for a novice to learn to use a product	Cognitive Walkthrough Formal Design Analysis	Concurrent Think-Aloud Video Analysis

Conclusion

To ensure maximum benefit, usability evaluation should be considered to be a dynamic process throughout the life cycle of the development of a product. Conducting evaluation both with and without end-users significantly improves the chances of ensuring a high degree of usability of the product for the end user.

References

Balbo, S. (1995). Software tools for evaluating the usability of user interfaces. In Anzai, Y., Ogawa, K. and Mori, H. (Eds). *Symbiosis of Human Artefact*, Volume 20B. Amsterdam: Elsevier Science B. V. 337-342.

Bevan, N. (1995a). Human-computer interaction standards. In Anzai, Y., Ogawa, K. and Mori, H. (Eds). *Symbiosis of Human Artefact*, Volume 20B. Amsterdam: Elsevier Science B. V., 885-890.

Bevan, N. (1995b). Human-computer interaction standards. In Anzai, Y., Ogawa, K. and Mori, H. (Eds). *Symbiosis of Human Artefact*, Volume 20B. Amsterdam: Elsevier Science B. V., 349-354.

Chignell, M. H., Motoyama, T. and Melo, V. (1995). Discount video analysis for usability engineering. In Anzai, Y., Ogawa, K. and Mori, H. (Eds). *Symbiosis of Human Artefact*, Volume 20B. Amsterdam: Elsevier Science B. V., 323-328.

Eberts, R. E. (1994). *User interface design*. Englewood Cliffs, NJ.: Prentice-Hall International, Inc.

Egan, D.E. (1988). Individual differences in human-computer interaction. In Helander, M. (Ed.). *Handbook of human-computer interaction*. Amsterdam: Elsevier Science B. V., 543-568.

Gould, J. D. (1988). How to design usable systems. In Helander, M. (Ed.). *Handbook of human-computer interaction*. Amsterdam: Elsevier Science B. V., 757-789.

Hix, D. (1995). Usability evaluation: How does it relate to software engineering?. In Anzai, Y., Ogawa, K. and Mori, H. (Eds). *Symbiosis of Human Artefact*, Volume 20B. Amsterdam: Elsevier Science B. V., 355-360.

Karat, J. (1988). Software evaluation methodologies. In Helander, M. (Ed.). *Handbook of human-computer interaction*. Amsterdam: Elsevier Science B. V., 891-903.

Kelley, T. and Allender, L. (1995). Why choose? A process approach to usability testing. In Anzai, Y., Ogawa, K. and Mori, H. (Eds). *Symbiosis of Human Artefact*, Volume 20B. Amsterdam: Elsevier Science B. V., 393-398.

Lee, N. S. and Park, J. H. (1995). Usability testing for a tele-radiology workstation. In Anzai, Y., Ogawa, K. and Mori, H. (Eds). *Symbiosis of Human Artefact*, Volume 20A. Amsterdam: Elsevier Science B. V., 1141-1146.

Lewis, C. and Rieman, J. (1994). *Task-centred interface design*. Shareware.

McManus, B. and Hammond, J. (1991). How to make usability work in the real world. In Hammond, J. H., Hall, R. R. and Kaplan, I. *OZCHI91: Australian CHISIG Conference Proceedings*, 97-102.

Nielsen, J. and Mack, R. L. (Eds). (1994). *Usability inspection methods*. NY.: John Wiley & Sons, Inc.

Perlman, G. (1988). Software tools for user interface development. In Helander, M. (Ed.). *Handbook of human-computer interaction*. Amsterdam: Elsevier Science B. V., 819-833.

Rasmussen, J. and Goodstein, L. P. (1988). Information technology and work. In Helander, M. (Ed.). *Handbook of human-computer interaction*. Amsterdam: Elsevier Science B. V., 175-181.

Rantanen, J. (1992). Usability testing of a scheduling system: How different usability testing methods support redesign of the user interface. In Rees, M. J. and Iannella, R. (Eds). *OZCHI92: Australian CHISIG Conference Proceedings*, 36-43.

Ravden, S. J. and Johnson, G. I. (1988). *Evaluating usability of human-computer interfaces: A practical method*. NY: John Wiley & Sons.

Reiterer, H. and Opperman, R. (1995). Standards and software-ergonomic evaluation. In Anzai, Y., Ogawa, K. and Mori, H. (Eds). *Symbiosis of Human Artefact*, Volume 20B. Amsterdam: Elsevier Science B. V., 361-366.

Stanney, K. and Mollaghasemi, M. (1995). A composite measure of usability for human-computer interface designs. In Anzai, Y., Ogawa, K. and Mori, H. (Eds). *Symbiosis of Human Artefact*, Volume 20B. Amsterdam: Elsevier Science B. V., 387-392.

Vora, P. R. and Helander, M. G. (1995). A Teaching method as an alternative to the concurrent think-aloud method for usability testing. In Anzai, Y., Ogawa, K. and Mori, H. (Eds). *Symbiosis of Human Artefact*, Volume 20B. Amsterdam: Elsevier Science B. V., 375-380.

Wilson, J. and Rosenberg, D. (1988). Rapid prototyping for user interface design. In Helander, M. (Ed.). *Handbook of human-computer interaction*. Amsterdam: Elsevier Science B. V., 859-875.

Whiteside, J., Bennett, J. and Holtzblatt, K. (1988). Usability engineering: Our experience and evolution. In Helander, M. (Ed.). *Handbook of human-computer interaction*. Amsterdam: Elsevier Science B. V., 791-817.

Ziegler, J. and Burmester, M. (1995). Structured human interface validation technique - SHIVA. In Anzai, Y., Ogawa, K. and Mori, H. (Eds). *Symbiosis of Human Artefact*, Volume 20B. Amsterdam: Elsevier Science B. V., 899-906.

Note

1. In this paper the term 'product' is used to encompass both products and systems.

Merle Conyer works as a user Performance Support Project Manager for Digital Equipment Corporation, on the Optus account. Merle is an experienced designer of interactive multimedia training courses and performance support solutions. Her academic background includes communication management, applied psychology, mathematics, education and instructional design. Phone (02) 342 1013; Fax: (02) 342 1055; Email: merle_conyer@yes.optus.com.au