Association for Information Systems
AIS Electronic Library (AISeL)

PACIS 1995 Proceedings

Pacific Asia Conference on Information Systems (PACIS)

December 1995

Intergrating Human Factors and Software Engineering Evaluations: An Illustratio with Reference to A Military Planning System

M. Colbert
University College London

Junsheng Long
University College London

J. Dowell
University College London

INTEGRATING HUMAN FACTORS AND SOFTWARE ENGINEERING EVALUATIONS: AN ILLUSTRATION WITH REFERENCE TO A MILITARY PLANNING SYSTEM

M. Colbert, J. Long and J. Dowell
University College London, UK

Evaluations of interactive human-computer systems and their effectiveness are conventionally considered from two different perspectives - a user-centred perspective and a computer-centred perspective. The weakness of this approach is that the two perspectives are often incommensurate. An alternative approach is to consider a system and its effectiveness from a single, integrated perspective, so reducing the risk of incommensuracy. This paper illustrates such an integrated approach to evaluation. The approach illustrated also encourages completeness of evaluation by conceiving the system and its effectiveness as an instance of a class of system and effectiveness. The system evaluated is a laboratory military planning system.

1 Introduction

1.1 Incommensurate Perspectives in Evaluation

Human-Computer Interaction (HCI) is an emergent discipline intended to comprise Human Factors (HF) and Software Engineering (SE) (Long & Dowell, 1989). A principal practice of HCI is evaluation. Following Whitefield, Wilson and Dowell (1991), evaluation may be defined as 'assessing the conformity between a system's [actual] effectiveness and its desired effectiveness'. Conventionally, the problem of late, summative HCI evaluation is conceived from two different perspectives - a human (user-centred) perspective and a computer (device-centred) perspective. Accordingly, two different aspects of system effectiveness are identified - usability and software quality. Consequently, two evaluations of the system are conducted and separately - an HF evaluation and an SE evaluation. The HF evaluation adopts a user-centred perspective and assesses usability (Whiteside, Bennett and Holzblatt, 1988). The SE evaluation adopts a device-centred perspective and assesses software quality (Henry and Kafura, 1981).

A strength of this approach is that it encourages complete evaluations by applying multiple perspectives to evaluation problems. A weakness is that user-centred and device-centred perspectives are often incommensurate. That is, it is difficult to translate between the perspectives. For example, the terms 'usability' and 're-usability' have different connotations within the two perspectives (see Boehm, Brown, Kaspar, Lipau, MacLeod and Merritt, 1978; Shackel, 1986). Incommensurate perspectives may adversely affect evaluation. For example, the outputs of HF and SE evaluations may be conceptually incompatible, or communicated poorly, and so may lead to misunderstanding between evaluators. Also, trading off the different implications of HF and SE evaluations for any re-design may be difficult or impossible to effect with coherence. As a result, resources are consumed guarding against, and coping with, the possible adverse effects of incommensurate perspectives - extended meetings of evaluators attempting to share perspectives, failing to form single coherent design teams (Gould, 1987), etc. There is, thus, interest in approaches reducing the risk and cost of incommensurate evaluations. One such approach changes the way in which evaluation problems are viewed, addressing them from a single, integrated perspective.

Such an approach is illustrated in this paper. The evaluation is conceived from a domain-oriented, systems perspective. Issues conventionally conceived as separate HF or SE issues are addressed similarly and together with respect to the same domain of application.

1.2 Encouraging Complete Evaluation

The objective of the present research is to contribute to *more effective* evaluation practice. In addition to addressing incommensuracy, the approach also attempts to retain the strength of current practice (the encouragement of completeness).

The present approach encourages completeness in two ways. First, evaluation is conceived from a single perspective that reflects the *integration* of user- and device-centred perspectives, within which it is possible to express HF and SE concerns. Second, a means of assessing the completeness of evaluation is provided. Specifically, the system to be evaluated and its effectiveness are conceived *as an instance of a class* of system and effectiveness. Thus, claims about completeness of an instance may be assessed with respect to the class (as will be seen later).

To avoid incommensurate evaluations and to encourage their completeness, this paper proposes an integrated approach that addresses instances of classes of evaluation, termed 'performance evaluation' (Denley & Long, 1989). The term derives from Dowell and Long's conception for HCI (1989), which reflects the authors' unified, domain-oriented and systems perspective. Dowell and Long's conception, thus, supports the integration of HF and SE perspectives and provides the basis for classes of evaluation problem. In the conception, effectiveness is expressed as performance (see Section 2.1).

The system used to illustrate the approach is one in which students interact with a laboratory prototype to produce plans for the off-load of men and equipment during amphibious military operations. The evaluation itself is intended as an illustration only, nevertheless one which demonstrates the concept of the approach.

The specific evaluation - that of 'students interacting with a prototype planning aid in the laboratory and its effectiveness' - is conceived as an instance of a class of evaluation - that of 'planners interacting with planning aids and their effectiveness'. This class is in turn conceived as

belonging to the class of 'humans interacting with computers and their effectiveness'. The system evaluated is University College London's Off-Load Planning System (OPS1). Desired performance for OPS1 is specified, then actual performance of OPS1 is obtained empirically and the conformance between desired and actual performance is assessed. Finally, the paper considers whether the intended integration of an HF, user-centred perspective and an SE, device-centred perspective has been achieved and whether the completeness of the evaluation has been encouraged.

2 The System to be Evaluated and its Effectiveness

2.1 Humans Interacting with Computers and Performance

Dowell and Long's conception (1989) expresses the general problem of HCI, that of systems of humans and computers interacting to perform work and their effectiveness. The conception, thus, offers the superordinate class for evaluation.

In the conception, an application domain (of a worksystem) is where work originates, is performed and has its consequences. It comprises one or more objects constituted of attributes (which have values). Task goals express a requirement for change in the value of these attributes, and goals are allocated to worksystems by organisations. A domain is distinct from, and delimits, a worksystem. A worksystem comprises at least two separate, but interacting sub-systems - of human behaviours interacting with computer behaviours. These human and computer behaviours are supported by mutually exclusive human and computer structures, and are executed to perform tasks effectively.

Effectiveness is expressed by the concept of performance, that is, how well a worksystem achieves its goals ('task quality'), and the costs that are incurred in so doing ('system costs'). Costs are incurred both by the human and the computer and are structural and behavioural.

Humans interacting with computers to perform work effectively, then, constitutes the superordinate class for evaluations.

2.2 Planners Interacting with Planning Aids and Planning Performance

Plans (documents), their potential for change and its realisation constitutes a class of work - planning work. Human planners interacting with computer-based planning aids constitute a class of worksystem - planning worksystems - whose superordinate class is humans interacting with computers to perform work effectively.

The domain of plans comprises a single type of object - plans. A plan, here, is a representation of goals and/or procedures for the work of controlling operations and has at least five attributes - availability, content, scope, view, and effect. *Availability* expresses the opportunity for individuals to access the plan and to put it to use, and may take the value 'when complete'. *Content* is what is specified, that is, the goal states and/or procedures, and may take the value 'on land by 06:00 hrs.' *Scope* defines and delimits the plan. It identifies the domain objects and the period of time for which the plan may specify goal states, and may take the value 'all units for the first day'. *View* is the type of language or representational scheme by which content is communicated, and may take the value 'table' or 'gantt chart'. *Effect* is the plan's added value, that is, the difference between control work (military operations), performed with and without the plan, and may take the value 'more fire-power'.

Planning tasks are required changes in availability, content, scope, view and effect. For example, a planning task may require the production 'by noon today (availability) of a table and illustrative diagram (view) showing the movement (content) of 152 Squadron for tomorrow's attack (scope) that ensures surprise will be achieved (effect)'.

Given this domain analysis, it is possible to describe the interactive planning behaviours of planning worksystems - one for each task goal. That is, planning worksystems: make plans available; consider the content of plans; scope plans; produce different views of plans; and consider the effect of plans. A planning worksystem's structures may be similarly specified. That is, planning worksystems possess structures for: making plans available; considering the content of plans; scoping plans; producing different views of plans; and considering the effect of plans.

Dimensions of plan quality are likewise based on domain analysis - one dimension of quality for each task goal. That is, a good plan is: appropriately available; well-scoped; may be viewed in acceptable ways; and has desirable content and effect. Sub-divisions of cost incurred by planning worksystems may also be based on the domain analysis. In addition to the distinction between human and computer costs, and behavioural and structural costs, costs for each task goal may be identified. For example, human structural costs may be divided into those incurred by the intention of: making the plan available; scoping; producing views; considering the content; and considering the effect.

Planners interacting with planning aids, then, constitutes a class for evaluations.

2.3 Students Interacting With an Off-Load Planning Workystem at University College London and Off-Load Planning Performance

The production of plans at University College London (UCL) for off-loading men and equipment during amphibious military operations is an instance of planning work; specifically, laboratory off-load planning work. Student subjects interacting with a prototype off-load planning worksystem at UCL constitutes an instance of a planning worksystem which is, in turn, an instance of a worksystem.

The laboratory plans produced are for off-loading men and equipment from landing ships (see upper window of Figure 1). The plan specifies who is to go ashore, where and when, and what is to transport them. *Availability* concerns how much of the plan has been produced by a deadline. *Content* concerns the goals that have been set for movement of assault craft, and lift and cohesion of the landing force. *Scope* refers to the assault craft and the landing force for which goals have been set. *View* refers to

File Edit Go Tools Objects Font Style

Off-Load Plan

Load No.	Contents	No. People	No. Vehicles	Desired Tactical Order	From	To	Timing Depart	Timing Land	Means
1	40COY/001	8	0	4	SHIP 1	BEACH 1	06:30	06:50	LCP 1
	40COY/002	8	0	5	SHIP 1	BEACH 1	06:30	06:50	LCP 1
	40COY/003	8	0	6	SHIP 1	BEACH 1	06:30	06:50	LCP 1
2	40COY/004	7	0	7	SHIP 1	BEACH 1	06:30	06:50	LCP 2
	40COY/005	4	0	8	SHIP 1	BEACH 1	06:30	06:50	LCP 2
	40COY/006	7	0	9	SHIP 1	BEACH 1	06:30	06:50	LCP 2
	40COY/007	4	0	10	SHIP 1	BEACH 1	06:30	06:50	LCP 2

Next Load - Pending

Load No.	Contents	No. People	No. Vehicles	Desired Tactical Order	From	To	Timing Depart	Timing Land	Means
3	40COY/008	7	0	11	SHIP 1	BEACH 2	06:30	06:50	LCP 10
	40COY/009	8	0	12	SHIP 1	BEACH 2	06:30	06:50	LCP 10
	40COY/010	7	0	13	SHIP 1	BEACH 2	06:30	06:50	LCP 10
	40COY/011	8	0	14	SHIP 1	BEACH 2	06:30	06:50	LCP 10

Clerical Check

Show Option: 1 | 2 | 3 | 4 | 5

Approve

Next Load - Options

Load No.
3

Option No.	Assessment	Contents (order no.)	Means	Total People	Total Vehicles
1.	OK	11,12,13,	LCP 3	26	0
2.	Overloaded	11,12,13,14	LCP 3	30	0
3.	Overloaded	11,12,13,	LCP 4	26	0
4.	Not Desired Order	16,17,18,19,	LCP 1	20	1
5.	Not Desired Order	15,16,17,18,	LCP 3	26	0

Next Load - Assessments

Criteria	Option 2
Power :	OK
Lift :	OK
Safety :	Overloaded
Cohesion :	OK
Fatigue :	OK
Total :	Overloaded

Assess | Compare | All Scores | Clear Away

Figure 1. OPS1.

the display of the plan as a table. Lastly, the *effect* of the plan concerns the improved movement, cohesion and lift of the landing force.

The task requires an initial off-load plan to be modified, and is described in more detail along with performance later (Section 3.3).

The laboratory Off-load Planning Worksystem (OPS1), comprises a HyperCard planning aid and student subjects, whose sole justification is to demonstrate the concept of integrated HF and SE. Consequently, some planning aid behaviours are simulated, rather than implemented, and simpler than those of an actual off-load system. Since it only exhibits *some* of the behaviours of the class of planning systems, OPS1 should be regarded as explicitly incomplete.

In OPS1, off-load plans are *made available* (added to the off-load table) one line at a time, from the beginning of the landing. A student subject 'approves' one of the load options offered and the aid adds the text in the 'Next Load - Pending' window to the bottom of the plan (see Figure 1). As part of *considering content*, the computer generates and displays five alternative next loads in the 'Next Load - Options' window. Details of any option are displayed on request in the 'Next Load - Pending' window. OPS1 also assesses the option in different ways and against a set of criteria. A subject selects the type of assessment via a button in the 'Next Load - Assessment' window. To discourage subjects from simply approving loads suggested by the computer, 'bugs' were introduced into the load assessment algorithm. For example, an overloaded craft might be erroneously rated highly by OPS1. A subject has to check OPS1's reasoning and adjust the load options scores to compensate for the bugs. A formatted Notepad helps subjects check systematically. Similarly, subjects actively consider the *scope* of the plan by locating and correcting bugs introduced by the assault craft and landing force selection algorithm; in this activity they are assisted by a 'clerical check' facility. In OPS1, only a single view of a plan is provided - as a table - so viewing behaviour in practice is not possible. Also, OPS1 does not consider the consequences of the plan for any hypothetical future landings, so may not be said to consider effect. OPS1 is to be regarded as an explicitly incomplete planning system.

OPS1's structures (the student subjects' mental representations and processes and the planning aid's stacks and scripts) may be similarly distinguished as structures for making off-load plans available, considering their content, and scoping plans. For example, the knowledge acquired during training about OPS1, amphibious operations and off-load planning, constitute human structures for considering content. The HyperCard stacks and scripts, that simulate an option generation algorithm, constitute a computer structure for considering content.

This section concludes the conceptualisation of the system to be evaluated and its effectiveness, as an instance of classes of system and effectiveness. The conceptualisation offers the potential for integrated HF and SE evaluation and so for avoiding the problem of multiple, incommensurate perspectives. In addition, it encourages completeness, because the instance may be verified with respect to the class. The evaluation is illustrated in the following section.

3 An Illustration of Integrated Evaluation: an Evaluation of the Performance of OPS1

This section begins with a consideration of the selectivity of the evaluation. Next, the procedure by which statements of performance were obtained is outlined. Then, the desired performance of OPS1 is specified and its actual performance obtained. Finally, the conformance of desired and actual performance of OPS1 is considered. Given the nature of this paper, an illustration of an approach involving a laboratory system, some conventional aspects of evaluation (choice of metrics; data analysis; etc.) are omitted.

3.1 Reasoning About the Completeness of the Evaluation

Only some aspects of OPS1 were evaluated. Selection was determined by the requirements to demonstrate the concept of integrated evaluation. Aspects of performance selected are: plan quality - availability, scope and content (plan content is taken to concern lift only); user and computer behavioural costs associated with considering content; and overall user and computer structural and behavioural costs. (Overall, here, means over all behaviours). Although the evaluation is incomplete, the selectivity being by intent, explicit and well-specified, does not jeopardise the concept demonstration of integrated evaluation.

3.2 Procedure for Obtaining Statements of Performance

3.2.1 Procedure

The actual performance of OPS1 was obtained by observing five student subjects learning and subsequently using the planning aid. Training required subjects to: read background material; watch demonstrations; explore OPS1; and complete multiple-choice tests, assessing what they had learnt. Following training, subjects were asked to produce two practice lines/loads for an off-load plan. Then, each subject attempted to produce five more lines/loads of the off-load plan within 50 minutes. Student subjects were unobtrusively observed and informally debriefed.

3.2.2 Metrics

The indices of performance used in the evaluation appear in Table 1 associated with their appropriate concept. The *quality of availability* was indexed by the mean percentage of lines/loads of the plan completed by the deadline. The *quality of the content* was indexed by the planned rate of lift, that is, the rate at which men and equipment were to be off-loaded, in terms of men per hour. The *quality of the scope* was indexed by errors concerning the landing force or assault craft, that is, confusions and inaccuracies in Columns 2-4 inclusive and Column 10 of the plan.

Concept	Index	Performance OPS0.5
Quality of Laboratory Off-Load Plans		
Availability	mean percentage of plan completed by the deadline	100%
Content	mean planned rate of lift: men/hr	278
Scope	mean errors of columns 2, 3, 4 and 10 of plan	1.8
Costs Incurred by Subjects		
Structural (Overall)	mean duration of exploration: mins, secs.	35' 39"
	mean correct answers on test	15.9
Behavioural (Overall)	mean workload rating	3.3
Behavioural (Considering Content)	mean notepad entries for the last two lines of the plan	11.2
Laboratory Costs Incurred by Off-Load Planning Aids		
Structural (Overall)	lines of code interface objects handlers that call handlers on other stacks	167 22 2
Behavioural (Overall)	mean time to produce a plan: mins, secs.	42' 54"
Behavioural (Considering Content)	estimated run time (standard interaction: secs.	2.5"

Table 1. Performace concepts and indices for the evaluation of Laboratory Off-Load Planning Worksystems and actual performance of OPS0.5.

Overall user behavioural costs were indexed by the mean workload rating by the students on a scale from 1 to 5. *User behavioural costs* associated with considering content were indexed by the mean number of entries in the 'Notepad'. (It is assumed that when considering content, subjects made an entry in the Notepad (Section 2.3).

Overall behavioural computer costs were indexed by the time taken to produce a five line plan. (The computer is assumed to be exhibiting behaviour constantly, even if only maintaining a display. Costs are assumed proportional to the time the structures supporting this behaviour are active.) *Computer behavioural costs associated with considering content* were indexed by the estimated run time of the show-option and assessment scripts. (Subjects are assumed to examine in detail, and view the aid's assessment of every load considered in the Notepad.)

Overall user structural costs were indexed by the mean number of correct answers on the training multiple-choice tests and the average length of time subjects spent exploring the aid. (Subjects are assumed to know nothing about off-load planning and the device prior to the study and that structural costs are incurred at a constant rate during exploration.)

Overall computer structural costs were indexed by the lines of code in the HyperCard scripts, the number of interface objects and the separability of different parts of the program (specifically, the number of handlers that call handlers located on other stacks). (It is assumed that smaller, more modular programs are easier for programmers to read and to write.)

These concepts, indices and the associated assumptions were considered sufficient to support the illustration of an integrated HF and SE evaluation in which completeness is encouraged.

Concept	Index	Performance Desired	of OPS1 Actual
Quality of Laboratory Off-Load Plans			
Availability	mean percentage of plan completed by the deadline	100%	91.5%
Content	mean planned rate of lift: men/hr	255-275	267
Scope	mean errors of columns 2, 3, 4 and 10 of plan	<1.8	0.4
Costs Incurred by Subjects			
Overall Structural	mean duration of exploration: mins, secs.	35' 39"	32' 31"
	mean correct answers on test	15.9	15.9
Overall Behavioural	mean workload rating	<3.3	3.0
Behavioural (Considering Content)	mean notepad entries for the last two lines of the plan	4 to 8	7.7
Laboratory Costs Incurred by Off-Load Planning Aids			
Structural	lines of code	167+	199
	interface objects	22+	40
	handlers that call handlers on other stacks	15+	1
Behavioural	mean time to produce a plan: mins, secs.	<42' 54"	40' 6"
Behavioural (Considering Content)	estimated run time (standard interaction: secs.	<2.5"	1.5"

Table 2. Desired and actual performance of OPS1

3.3 Desired Performance of OPS1

The desired performance of OPS1 (Table 2) is simply asserted, for present purposes, sometimes in absolute terms and sometimes relative to a previous version of OPS1, that is OPS0.5 (Table 1). First, a seven line plan must be available 50 minutes after the start of the task. (Other individuals are assumed to need the plan by this time.) Second, the plan must specify lift at a rate between 255 and 275 men/hr (assumed necessary to achieve military superiority). With OPS0.5, subjects achieved a lift of 278 men/hr. (Table 1). Third, the scope of the plan must be better than that of OPS0.5, i.e. fewer than 1.8 clerical errors. Fourth, overall structural and behavioural user costs must be less than those of OPS0.5, that is, not more than 35mins 39 secs to explore the system, at least 15.9 correct answers on the test and a mean workload rating of not more than 3.3, respectively. Fifth, user behavioural costs associated with considering content should be low - four to six entries in the Notepad. (Given the aid's bugs, six entries is the minimum required to check OPS1 explicitly). Sixth, provided overall computer behavioural costs are less than those incurred by OPS0.5, that is, a task is completed in less than 42mins 54 secs, and other performance criteria are satisfied, overall structural computer costs may be slightly more than those incurred by OPS0.5 (a more effective interaction may require a larger program). Finally, computer behavioural costs associated with considering content should be low - less than 2.5 secs for a standard interaction, that is, one in which a subject examines each alternative load and views the aid's assessment. A longer time increases overall user costs because subjects become frustrated and complain that such delays interrupt their train of thought.

3.4 The Conformance Between Actual and Desired Performance of OPS1

The conformance between OPS1's desired and actual performance (Table 2) is summarised below.

3.4.1 Task Quality

The availability quality of OPS1's plan is unacceptable, since, on average, only 6.4 lines 91.5% were available by the deadline. OPS1's considering content and making

available behaviours are, with respect to availability, risky. They fail to guard against an unavailable (late) plan. If an availability goal is expressed as a deadline, and a plan is produced one load at a time, then whenever a subject limits the rate of incurring costs (works too slowly), the result is likely to be an unavailable (late) plan. Alternative behaviours may make an unavailable plan less likely. For example, suppose the computer considers the content of a complete plan, and makes it available immediately. Then, a subject could consider content by selecting any load and, if necessary, replacing it. A complete draft could then be checked by the computer, and any loads affected by the subject's modification could be revised. With such behaviours, if a subject works too slowly, the result is likely to be an available plan with sub-optimal content (some loads may not have been considered by the deadline), rather than an unavailable plan (included loads are optimal, but some loads are not included).

The scope of the plan is acceptable. Indeed, it is better than desired - fewer clerical errors are made with OPS1 than with OPS0.5. Better scoping accounts for approximately 50% of the improvement, and only increases computer costs by two interface objects and 25 lines of code.

The content of the plan is lower than that of OPS0.5, but still higher than desired. There appear to be two reasons: (i) subjects still consider content inappropriately, sometimes failing to identify overloaded craft. (Subjects had not yet acquired the structures (representations of domain knowledge) to consider content); and (ii) subjects' 'making available' behaviours are not executed appropriately.

3.4.2 User and Computer Behavioural Costs Associated with Considering Content

User behavioural costs associated with considering content are acceptable. The display of all scores of all loads helped subjects to consider content rapidly and accurately. It also helped them to learn how to consider content during the practice session. This display incurs relatively few computer costs (one interface object and five lines of code), but evidently enables subjects to consider content systematically and economically (Section 4). Estimated computer behavioural costs associated with considering content are comparable to those incurred by OPS0.5, and so acceptable.

3.4.3. Overall User and Computer Behavioural and Structural Costs

Overall user behavioural and structural costs are acceptable, relative to those of OPS0.5. Subjects scored the same on the training multiple-choice tests. Overall computer behavioural costs are acceptable. Subjects took a similar amount of time to explore OPS1 as OPS0.5 and subjective workload ratings are also similar. Overall computer structural costs have increased to an unacceptable level, however, and suggest a waste of computer resources. The computer's translation and comparison behaviours incur considerable costs (an additional 9 interface objects and 50 lines of code), but have little impact on task quality. OPS1 appears to have some excessive functionality.

In summary, OPS1 has only partially achieved its desired level of performance. Performance is as desired with respect to scope, overall user structural and behavioural costs, and user behavioural costs associated with considering content. However, its performance with respect to plan content, plan availability and overall computer structural costs is unacceptable.

The illustration of performance evaluation is now concluded. The following section considers the potential of performance evaluation as an alternative to, and a progression from, conventional evaluation.

4. Discussion

This evaluation constitutes an integration of evaluation from both an HF, user-centred perspective and from an SE, device-centred perspective. Adopting a domain-oriented, systems perspective, issues conventionally conceived as HF and SE, and considered within two different perspectives, have been unified as HCI issues within a single evaluation. For example, the evaluation considered HF issues, such as OPS1's usability, learnability and utility. In conventional terms, some types of assessment (all the scores of all the options) were easier to access with OPS1, and so learning was more systematic and effective. This outcome affected the utility of OPS1 because, when subjects had to plan under time pressure, they were more able to plan efficiently, and correctly calculate the best load. This issue was expressed as an improvement in the quality of plan content due to a reduction in user behavioural costs associated with considering content, and for a minimal increase in computer structural costs. (Note that the trade-off between usability, learnability and utility, and the implications for the computer are explicit in the integrated evaluation.)

Also considered were SE issues, such as OPS1's modularity and functionality. In conventional terms, the isolation and independence of parts of the program were increased by the separation of the 'assessment' and 'detailed option display' scripts in OPS1 compared to OPS0.5. The evaluation also questioned the functionality of the translation and comparison scripts, because these facilities were rarely used. These issues were expressed as a reduction in computer behavioural costs, incurred when considering content and an increase in overall computer structural costs for little impact on plan content. (Note that the trade-off between computer costs and impact of plan quality is explicit in the integrated evaluation.)

The evaluation also addressed issues conventionally not identified as either HF or SE issues, but which evaluation should expect either or both to address. For example, conventionally, requiring a plan to be produced one line at a time and to be available by a deadline risks the production of an unavailable (late) plan. A preferable alternative may be the production of a complete draft plan early on, and the refinement of the draft. In performance evaluation, such issues were expressed as alternative implementations of 'making available and considering content' behaviours and their relative advantages for achieving certain kinds of availability goals.

The integrated performance evaluation, then, addressed some of the issues that, conventionally, would be considered from user- and device-centred perspectives and within two different evaluations, from a single, domain-oriented systems perspective and within a single evaluation. As such, integrated performance evaluation would seem to have the potential to reduce the adverse effects of incommmensurate perspectives - poor communication, duplication, ommission and delay.

The evaluation also supported reasoning about the completeness of evaluation. It was possible to specify the completeness of the evaluation in terms of how many aspects of performance implicated by the conceptualisation of planning as a general class were addressed in the instance of off-load planning in the laboratory. For example, this evaluation intentionally did not consider how well the off-load plan was represented, or the extent to which off-load plans achieved their desired effect on (hypothetical) landings. But it did consider at least one aspect of the availability, scope and content of off-load plans. The evaluation, then, could claim to be complete with respect to those aspects selected. Had all aspects been selected, the evaluation could have been verified as complete (with respect to planning systems and their effectiveness as a class). Further, the fore-going evaluation did not identify additional issues which required the conceptualisation of planning systems and their effectiveness as a class to be modified. As such, performance evaluation also has the potential to reduce the risk of incomplete assessments of system effectiveness.

Acknowledgement

This research was funded by UK MoD (2047/148) in association with DRA Portsdown. The views are the authors' and do not necessarily reflect those of H.M. Govt. Special thanks to Christine Busittil for her data analysis.

References

Boehm, B. W., Brown, J. R., Kaspar, H., Lipow, M., MacLeod, G. and Merritt, M. "Characteristics of Software Quality," *TRW Series of Software Technology*, North-Holland, Amsterdam, Netherlands, 1978.

Denley, I. and Long, J. "A Framework for Evaluation Practice," in *Contemporary Ergonomics 1990*, Proceedings of the Ergonomics Society's Annual Conference, E. J. Lovesey (ed.), Taylor & Francis, London, UK, 1990, pp.31-37.

Dowell, J. and Long, J. "Towards a Conception for an Engineering Discipline of Human Factors," *Ergonomics* (32:11), November 1989, pp.1513-1535.

Gould, J. D. "How to Design Usable Systems," *Proceedings of INTERACT '87*, H-J. Bullinger & B. Shackel (eds.), Elsevier, Amsterdam, Netherlands, 1987, pp.35-39.

Henry, S. and Kafura, D. "Software structure metrics based on information flow," *IEEE Trans. Software Eng.*, SE-7(5), 1981.

Long, J. and Dowell, J. "Conceptions for the Discipline of HCI: Craft, Applied Science and Engineering," *Proceedings of the Fifth Conference of the BCS HCI SIG*, A. Sutcliffe and L. Macaulay (eds), Cambridge University Press, Cambridge, UK, 1989, pp.9-23.

Shackel, B. "Ergonomics in Design for Usability," *Proceedings of the Second Conference of the BCS HCI SIG*, M. Harrison and A. Monk (eds), Cambridge University Press, Cambridge, UK, 1986, pp.44-64.

Whitefield, A., Wilson, F. and Dowell, J. "A Framework for Human Factors Evaluation," *Behaviour and Information Technology* (10:1), January 1991, pp.65-79.

Whiteside, J., Bennett, J. and Holzblatt, K. "Usability Engineering: Our Experience and Evaluation", *Handbook of Human-Computer Interaction*, M. Helander (ed.), Elsevier, Amsterdam, Netherlands, 1988, pp.791-817.