Reconditioned Merchandise: Extended Structured Report Formats in Usability Inspection

Gilbert Cockton, Alan Woolrych and Mark Hindmarch
School of Computing & Technology,
University of Sunderland,
PO Box 299 Sunderland SR6 0DD, UK
{Firstname.Lastname}@sunderland.ac.uk

ABSTRACT

Structured Problem Report Formats have been key to improving the assessment of usability methods. Once extended to record analysts' rationales, they not only reveal analyst behaviour but also change it. We report on two versions of an Extended Structured Report Format for usability problems, briefly noting their impact on analyst behaviour, but more extensively presenting insights into decision making during usability inspection, thus validating and refining a model of evaluation performance.

Author Keywords
Usability Evaluation, Usability Inspection

ACM Classification Keywords
ACM: H.5.2 – User Interfaces

INTRODUCTION

We describe how we have extended report formats in order to validate and refine a model of analyst performance for usability evaluation. We next summarize how applying research innovations to address Gray and Salzman's concerns about assessments of usability methods [4] let us infer a logical model of analyst performance during evaluation. We then describe how we have extended a key research instrument, the Structured Problem Report Format (SPRF), in order to validate this model, which explains and predicts performance on the basis of logically distinct phases of analyst activities during usability evaluation. The model takes its name from two key central phases of usability evaluation: problem discovery and problem analysis. We have observed how successful discovery and analysis of usability problems depends on effective use of a range of knowledge resources. The model is thus called the *DARe Model* to highlight the key role of **D**iscovery and **A**nalysis **Re**sources during usability evaluation.

DAMAGED MERCHANDISE: ASSESSMENTS OF USABILITY EVALUATION METHODS

In 1998, Gray and Salzman raised serious doubts about the quality of assessments and comparisons of evaluation methods [4], which they saw as *Damaged Merchandise*. They analyzed flaws in existing studies as breaching forms of validity: statistical conclusion, internal, construct, external, and conclusion validity.

We must improve the validity of evaluation method assessment. Some improvements simply require good practice: statistical conclusion validity needs good choice and interpretation of statistical tests; conclusion validity requires sound conclusions from correctly interpreted data.

Innovation is required to address the three other forms of validity. Internal validity depends on careful control of potential confounds. Construct validity requires careful experimental design, to ensure that intended causes and effects are actually measured. We need new instruments and procedures for both. External validity concerns generalization to the real world, which we cannot attempt until we have addressed internal and construct validity.

When assessing usability evaluation methods, all studies must correctly identify true and false positives, true and false negatives, and oversights, by analysing predictions from inspection against problems discovered in user testing. Common challenges are faced when tackling internal and construct validity. General tactics can be developed to improve validity for any method assessment.

This paper reports how a novel research instrument was transformed. What began as a simple control for confounds was extended to measure new constructs, but also improved analyst performance. We thus ask: can damaged merchandise can be reconditioned with appropriate tools?

RECONDITIONED MERCHANDISE: FIXING ASSESSMENTS OF USABILITY METHODS

Validity is a major challenge for usability inspection method (UIM) assessment. We must ensure that a false positive is not due to a flaw in method assessment. Similarly, an unpredicted problem must also be shown to be due to the UIM and not to the assessment. Confounding variables can easily bias a study. For example:

1. *Analyst misunderstanding* of the UIM can result in missed problems or inappropriate analysis of predicted problems (false negatives, false positives)

CHI 2004, April 24–29, 2004, Vienna, Austria.
ACM 1-58113-703-6/04/0004.

2. *Researcher error in merging predictions* — to produce a single set of predictions for each analyst (group) and for all analysts — can alter measures of (in)appropriate method usage, as well as corrupting problem counts
3. *Failure of user testing to expose a predicted problem* can result in incorrect scoping of an inspection method
4. *Researcher error in extracting/merging actual problems* — to produce one set of problems from user testing — combines risks for previous two confounds
5. *Researcher error in matching predicted to actual problems* adds further risks of misclassifying predictions as (un)confirmed.

The first potential confound is the most challenging, but can be partially addressed via common training materials, e.g. as in [2]. The third confound can be controlled within a falsification study, where features associated with predicted problems are systematically stressed in user testing, where possible by recreating the contexts in which problems are predicted to occur [3]. The fourth confound can be controlled via structured problem extraction [1].

The second and fifth potential confounds can both be reduced by using a usability problem report format [5]. In this paper, we report how an extended report format has improved our understanding of usability inspection.

DERIVING A MODEL OF EVALUATION PERFORMANCE

Scoping and assessing Heuristic Evaluation (HE) [2] let us reflect on why some problems get missed but others are falsely predicted. Missed problems are either never found or are mistakenly dropped. False positives get found, but are mistakenly preserved! Thus to explain analyst performance, we must distinguish *finding* and *oversight* of problems from *confirmation* and *elimination*.

This forms the basis for a very simple *phase model* of analyst performance in usability evaluation, which begins with preparation and ends with recommendation. *Discovery* and *analysis* are the two core phases, which we focus on to better understand evaluation performance. Thus missed problems in HE [6] were analyzed in terms of *discoverability*, leading to a highly significant result that problems were rarely found when they needed more than trivial inspection to uncover them [2]. Unlike such missed problems, false positives are found but not eliminated. To explain analyst performance, we thus study both discovery and analysis. Errors are possible in both. A problem can be missed during discovery, but it can equally well be incorrectly eliminated during analysis. Alternatively, an improbable problem considered during discovery can be incorrectly confirmed during analysis.

The DARe model was a logical inference from our first major study [2]. Our initial problem format [5] was designed to allow control of confounds both during merging of analyst problems and when matching predicted problems to the results of user testing. It eased the researcher's task, but focused on problem confirmation. We thus had no direct evidence of problem discovery or elimination. We thus extended our initial SPRF.

EXTENDING STRUCTURED REPORT FORMATS

Our first Extended SPRF (ESPRF) has four parts. The existing SPRF became Part 1, with its four elements [2]: Problem Description, Likely/Actual Difficulties, Specific Contexts, and Assumed Causes. The SPRF identifies the usability problem. Its four elements provide multiple points of reference for problem merging and matching.

Part 2 of the ESPRF addressed discovery resources and methods. Analysts had to explain their discovery and indicate if their approach was system- or user-centered and unstructured or structured. This yields four tactics: system scanning, system searching, goal playing and method following. The first two are system-centered, the first and third are unstructured. Different knowledge resources are required: little if any for system scanning; product knowledge for system searching; user/domain knowledge for goal playing; and task knowledge for method following.

Part 3 dealt specifically with heuristic application to individual problems. Analysts had to provide evidence of conformance rather than just name a heuristic. Part 4 required analysts to justify any problem elimination, with specific reference to user impact and behavior.

This format was used in a second unreported study. The priority was to replicate results from the pilot. We thus used the same ESPRF. Having replicated the pilot results, we later modified the ESPRF to address some remaining gaps in data collection: a lack of confirmation rationales and specific information on discovery knowledge resources. The initial extensions focused on information that was clearly missing from the SPRF, i.e., how analysts approach discovery and whether/why elimination occurs. We had not focused on confirmation of probable problems.

ESPRF v2 is identical to v1 except for Part 2, where we requested an explicit confirmation rationale and structure for the discovery explanation, as well as details of goals or task steps involved in user-centered problem discovery (to expose relevant knowledge resources). ESPRF (v2) is at http://www.cet.sunderland.ac.uk/~cs0gco/sesprf.doc

VALIDATION OF THE DARe MODEL

A pilot study [3], an unreported replication and a recent study using ESPRF (v2) provide direct evidence that good analyst performance in usability inspection is characterized by appropriate use of knowledge resources in distinct phases of discovery and analysis.

In all three studies, analyst groups carried out a HE [6] of a local transport web-site using ESPRF (v1). All came from a final year HCI course. Most had two years of HCI education. Many had extensive work experience in the IT industry, comparing well with typical recent graduate entrants to ICT.

The pilot was intended to identify the ESPRF's limits to identify data that must be gathered via observation of analysts and debriefing interviews. ESPRF was more effective than we expected, providing extensive direct evidence of discovery and analysis resources in action, as well as demonstrating how and why improbable problems were eliminated. This qualitative validation of the DARe model has not been reported before. We now present examples of discovery and analysis resources in action. Examples are from group reports in the pilot unless stated otherwise (1.3 means Prediction 3 from Group 1).

Discovery Resources

With ESPRF (v1), analysts indicated their discovery method and provide an unstructured explanation how they encountered a reported problem. We collected good examples of a range of discovery behaviors. User testing confirmed all predications associated with the examples.

System Scanning

Report 1.1 stated, "We scanned the website looking for problems." For too many analysts, this is the limit of forethought and planning during inspection.

System Searching

Report 2.2 stated: "we decided to search through the subsections systematically" resulting in discovery of a problem "within the metro section, second link on the navigation bar." Where groups were searching systematically, they could clearly articulate their approach.

Goal Playing

Report 1.5 used this unstructured user-centred approach: "we tried to purchase a ticket using an incorrect credit card number and the site gave no feedback about the incorrect card number." Selecting this goal requires a basic level of domain knowledge about credit cards, resulting in a well grounded valid prediction.

Method Following

"Firstly we tried to read the passenger charter by selecting the relevant link and secondly we tried to plan a journey using the journey planner option [finding] little or no feedback given regarding … system status" (Report 8.4).

Analysis Resources

We collected convincing examples of a wide range of confirmation and elimination behaviors that demonstrate use of key knowledge resources.

Models of Users

Knowledge of users can quickly confirm or eliminate possible problems: "no attempt is made to cater for foreign users … it assumes you can read English" (Report 3.4).

Models of Interaction

Failure to understand distributed cognition and how users learn within interaction allows confirmation of problems that underestimate human capabilities, e.g., "the user might get confused when looking at the site because of the change in [background] colour" (Report 1.1) — a false positive resulting from a slapdash discovery approach. No test users commented on changes in background colour, or were confused by it. Conversely, a site that gives "no feedback about the incorrect [credit] card number" (Report 1.5) fails to support learning. Users will be left ignorant of task failure for days if not weeks.

Models of Tasks/Activities

A well-grounded understanding of tasks and their critical parameters of outcome and execution is essential to focusing on important problems: "the difference in time from expected task duration to actual task duration was only a matter of seconds and even though the process was slightly frustrating, the relief at finding the info after a minute or two soon outweighed the disappointment". Report 2.4 demonstrates fair judgement of what matters: getting to information within reasonable, not record, time. A false negative was correctly avoided.

Knowledge of the Application Domain

Report 6.6 used domain knowledge when exploring the goal of travel to/from university: "the journey planner only offers one way to go … if air is excluded, and will only offer the metro as an option if the bus option is excluded". Basic domain knowledge was used here: journeys occur in both directions —— flying or not you can travel by train, and without having to get on a bus too!

Product Knowledge and Knowledge of Interaction Design

The role of these analysis resources was inferred from bogus problem reports in [2]. There were no bogus reports in the pilot or replication, indicating a beneficial practical side effect of analyst self-reporting via ESPRFs.

Technical Knowledge

"Due to lack of alt tags the user will not be able to determine where the links take them to [if they are] viewing with the graphics turned off on their own browser" (Report 2.6, Replication). A basic knowledge of browser options is needed to confirm this.

REFINING THE DARe MODEL

The ESPRF not only validated the DARe model, but refined it. For example, we encountered *Technical Knowledge* as a confirmation resource for the first time in the replication. We could not infer this resource from [2].

A surprising result from the pilot study has already been reported [3]. Analysts used heuristics more appropriately and predicted fewer false positives in comparison with the initial major study [2]. These results have now been replicated. As we made no claims for increased thoroughness, we did not need to carry out falsification tests on new predictions from the replication. Worst and best case validity scores are thus given: worst for when all new predictions are false positives;

best for when none are. Table 1 presents these results. Validity is a measure of accuracy (percentage of predictions confirmed). Appropriateness measures correct method use (percentage of heuristic applications that are correct).

	Validity	Appropriate
Initial Study [2]	31%	31%
Pilot [3]	50%	57%
Replication (Worst)	48%	60%
Replication (Best)	63%	60%

Table 1: Validity and Appropriateness

This surprise led us to examine data that were only available from the ESPRF. We could correlate discovery method use with elimination rates, validity and appropriateness of heuristic use. System scanning was associated with a lower rate of problem elimination, a higher rate of false positives and a lower rate of appropriateness [3]. In a nutshell, easily found tended to mean easily kept, easily mistaken and easily confused!

Most analysts who use an ESPRF appear to become aware of key aspects of the DARe model (discovery tactics, considering elimination) even when they apply system-centered discovery. The difference between system searchers and user-centered approaches is far less marked, than that between system scanners and others. This suggests that being systematic has a pay off that almost matches that of being user-centred.

The report format has also provided revealing evidence on how discovery and analysis resources interact to produce (un)successful predictions. Successful predictions often involve multiple resources. An analyst group that carefully confirmed a problem using domain knowledge (*Knowledge of the Application Domain* above) had discovered it using minimal effort: “After seeing the different looking button, I decided to click on it and explore further” (Report 6.6).

Using different resources for discovery and analysis appears to matter. Improbable problems found via goal or method based discovery are too readily confirmed by the same knowledge: “some users may not realise the buttons at the top of the screen are links as they may go to the buttons underneath first as they stand out more” which was discovered through the (vague) goal: “find out how much a ticket for a student is” (Report 2.3). Clear evidence of task success in a method description was bizarrely used to confirm the prediction: “once on students, a list of zone prices came on screen.”

In contrast, use of disjoint multiple resources across analysis and discovery appears to be more effective. Thus Report 2.4 eliminated a possible problem (redundant text/graphic links) by drawing on three *further* types of resource. A combination of both logos and text as links was seen to support a range of user knowledge, allowing learning in context (interaction knowledge resource) and consistent visual design (knowledge of design resource). These resources successfully countered the task (goal) knowledge used for problem discovery, correctly avoiding a false negative.

CONCLUSIONS

ESPRFs have let us demonstrate that successful usability analysts using HE apply several knowledge resources in distinct phases of discovery and analysis. In successful UIM use: most, if not all, problems are found (thoroughness); few, if any, false positives arise (validity); and UIMs are correctly applied to allow sound derivation of recommendations. We have evidence that poor usability analysts fail to eliminate improbable problems by overlooking key knowledge resources, or miss them by applying weak discovery tactics. Strong discovery tactics however can result in false positives if complementary analysis resources are not used.

ESPRFs are effective research instruments that support validation, refinement and extension of models of evaluation performance, in our case, the DARe model for UIMs. We have clear examples of specific knowledge resources in use, and have encountered unanticipated types of resource (e.g., technical knowledge) as a result.

Since ESPRFs embody the models that they seek to instantiate, they can guide analysts to more effective discovery methods and more thorough analysis of predictions. Surprisingly, they not only shed light on analyst performance, but can actually improve it. HCI needs more such fortunate fusions of theory and practice.

REFERENCES

[1] Cockton, G. and Lavery, D. "A Framework for Usability Problem Extraction”, in *Proc. INTERACT 99*, eds. A. Sasse and C. Johnson, 347-355, 1999

[2] Cockton, G. and Woolrych, A., “Understanding Inspection Methods: Lessons from an Assessment of Heuristic Evaluation,” in *People & Computers XV*, eds. A. Blandford & J. Vanderdonckt, 2001

[3] Cockton, G., Woolrych, A., Hall, L., and Hindmarch, M., “Changing Analysts’ Tunes: The Surprising Impact of a New Instrument for Usability Inspection Method Assessment?” *People and Computers XVII*. Springer-Verlag, 145-161, 2003

[4] Gray, W.D. & Salzman, M., “Damaged Merchandise? A Review of Experiments that Compare Usability Evaluation Methods”, *HCI*, 13(3), 203-261 1998

[5] Lavery, D. and Cockton, G., “Representing Predicted and Actual Usability Problems”, in *Proc. Int. Workshop on Representations in Interactive Software Development*, QMW London, 97-108, 1997

[6] Nielsen, J. “Enhancing the Explanatory Power of Usability Heuristics”, *Proc. CHI '94*, 152-158, 1994.