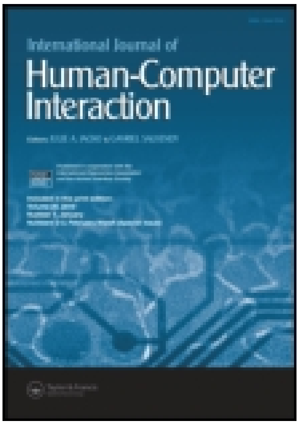

International Journal of Human-Computer Interaction
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/hihc20

Ingredients and Meals Rather Than Recipes: A Proposal for Research That Does Not Treat Usability Evaluation Methods as Indivisible Wholes

Alan Woolrych [a], Kasper Hornbæk [b], Erik Frøkjær [b] & Gilbert Cockton [c]

[a] University of Sunderland, Sunderland, UK

[b] University of Copenhagen, Copenhagen, Denmark

[c] Northumbria University, Newcastle upon Tyne, UK

Available online: 04 Mar 2011

To cite this article: Alan Woolrych, Kasper Hornbæk, Erik Frøkjær & Gilbert Cockton (2011): Ingredients and Meals Rather Than Recipes: A Proposal for Research That Does Not Treat Usability Evaluation Methods as Indivisible Wholes, International Journal of Human-Computer Interaction, 27:10, 940-970

To link to this article: http://dx.doi.org/10.1080/10447318.2011.555314

INTL. JOURNAL OF HUMAN–COMPUTER INTERACTION, *27*(10), 940–970, 2011
Copyright © Taylor & Francis Group, LLC
ISSN: 1044-7318 print / 1532-7590 online
DOI: 10.1080/10447318.2011.555314

Taylor & Francis
Taylor & Francis Group

Ingredients and Meals Rather Than Recipes: A Proposal for Research That Does Not Treat Usability Evaluation Methods as Indivisible Wholes

Alan Woolrych[1], Kasper Hornbæk[2], Erik Frøkjær[2], and Gilbert Cockton[3]
[1]University of Sunderland, Sunderland, UK
[2]University of Copenhagen, Copenhagen, Denmark
[3]Northumbria University, Newcastle upon Tyne, UK

To better support usability practice, most usability research focuses on evaluation methods. New ideas in usability research are mostly proposed as new evaluation methods. Many publications describe experiments that compare methods. Comparisons may indicate that some methods have important deficiencies, and thus often advise usability practitioners to prefer a specific method in a particular situation. An expectation persists in human–computer interaction (HCI) that results about evaluation methods should be the standard "unit of contribution" rather than favoring larger units (e.g., usability work as a whole) or smaller ones (e.g., the impact of specific aspects of a method). This article argues that these foci on comparisons and method innovations ignore the reality that usability evaluation methods are loose incomplete collections of resources, which successful practitioners configure, adapt, and complement to match specific project circumstances. Through a review of existing research on methods and resources, resources associated with specific evaluation methods, and ones that can complement existing methods, or be used separately, are identified. Next, a generic classification scheme for evaluation resources is developed, and the scheme is extended with project specific resources that impact the effective use of methods. With these reviews and analyses in place, implications for research, teaching, and practice are derived. Throughout, the article draws on culinary analogies. A recipe is nothing without its ingredients, and just as the quality of what is cooked reflects the quality of its ingredients, so too does the quality of usability work reflect the quality of resources as configured and combined. A method, like a recipe,

We thank members of MAUSE Working Group 2 for their contributions to the development of these analyses. In particular, Dominic Furness and Tobias Uldall-Espersen made contributions to parts of the WG2 report (Cockton & Woolrych, 2009) from which some presented material is derived. An archive of the MAUSE project, with all deliverables, can be found at http://www.cost294.org. The follow on COST project, TwinTide (http://twintide.org/), is focused on domain specific resources and requirements that influence the transferability of methods between different domains. Some of the presented material has benefitted from discussions with TwinTide members.

Correspondence should be addressed to Kasper Hornbæk, Department of Computer Science, University of Copenhagen, Universitetsparken 1, Copenhagen, DK-2100, Denmark. E-mail: kash@diku.dk

is at best a guide to action for those adopting approaches to usability that are new to them. As with culinary dishes, HCI needs to focus more on what gets cooked, and how it gets cooked, and not just on how recipes suggest that it could be cooked.

1. INTRODUCTION

Research that assesses usability evaluation methods has been in crisis for over a decade. There are fewer publications at conferences like ACM's conference on Human Factors in Computing Systems that assess methods. Highly critical reviews of research that assess usability evaluation methods and the advice generated from such research have appeared (Gray & Salzman, 1998; Hornbæk, 2010). Thus although human–computer interaction (HCI) evaluation practices need to evolve, perhaps radically, to be fit for purpose as the scope of Interaction Design expands, this does not appear to be happening (Barkhuus & Rode 2007). For Barkhuus and Rode (2007), there are risks that inadequate evaluation practices are becoming prematurely standardized. However, if the standard of research on evaluation method assessment cannot be improved, then more appropriate methods are unlikely to be disseminated through the research literature.

Our position is that this continuing crisis may be a result of how research on usability evaluation continues to focus on method, ignoring the realities of maturing usability practice. For example, it is now frequently observed that experts in practice adapt and mix methods (Gulliksen, Boivie, & Göransson, 2006; Rosenbaum, 2008). However, method assessment research still mostly treats usability evaluation methods as complete indivisible wholes, that is, fixed combinations of decisions about, for instance, the role of participants, of procedures for conducting the evaluation, and of the criteria used for determining whether something is a problem. Thus, Heuristic Evaluation (HE; Molich & Nielsen, 1990; Nielsen & Molich, 1990) is assumed to make choices of procedure for an evaluation, including strategies for discovering problems. Further, research comparing HE to other approaches (Bailey, Allan, & Raiello, 1992; Capra & Smith-Jackson, 2005; Hornbæk & Frøkjær, 2004) typically compares a set of choices for HE (many specific to the experimental setup) with a partially controllable set of choices for another evaluation approach. The assumption is that methods are being compared, when in reality it is evaluation settings. These will differ with regard not only to specific configurations of the methods being used (as these are treated as the independent variables in an experimental study) but also to other potentially confounding variables. To control such confounds, a host of studies would have to be run to test different method configurations in different evaluation settings. It is not clear how the results of such complex studies could be relevant to practitioners, even if they were practically possible.

This article argues that the conception of indivisible methods, and the research that arises from it, is problematic. We flesh out the argument as follows. Section 2 discusses what a method is and what may reasonably be expected from using one. Section 3 outlines our approach, looking not at methods but at resources and the work in which they are applied. After that, we present a list of ingredients and discuss research on their influence on evaluation (sections 4 and 5). Next, we argue the need to look more closely at usability work, in particular, how practitioners combine, and may be supported in combining, specific resources (section 6). Finally, section 7 presents implications of our view, in particular for research but also for teaching and practice, outlining potential benefits for both research and practice.

2. THE REALITIES OF METHOD USE

We need to reconsider the status of methods in HCI. HCI research that compares evaluation methods is motivated by misconceptions about the nature of method that are deeply rooted in centuries of Western thinking. Much of our hopes for methods date back to Descartes (Sorell, 2001) and the ideals laid out in his *Discourse on the Method of Rightly Conducting the Reason* (1637), within which he sought to outline a method for producing *certain* knowledge. Over intervening centuries, the ideal of a "scientific method" has been regularly discussed and refined but never perfected. Although the immense historical progress that arises from scientific endeavors is beyond doubt, no account of scientific method has been sufficiently robust to establish exactly why scientific endeavors succeed, or even what these must (not) entail (Okasha, 2002). Paul Feyerabend (1975) could thus survey a wide range of scientific breakthroughs and show that none resulted from following any explicit method, whether contemporary or formalized (centuries) later.

The belief that a scientific method can be followed diligently to ensure production of certain knowledge strongly influenced the design methods movement of the 1970s, but the author of the classic work in this area (Jones, 1992) came to disown its values and aspirations (Jones, 1992). A similar argument on the limited power of methods and rules has been made by Naur in other areas of human activity, such as language, musical composition, computing, and mathematical theorem proving (Naur, 1995), and by Stuart and Hubert Dreyfus in characterizing skill acquisition (Dreyfus & Dreyfus, 1986). Yet, despite these widespread understandings of the realities of method use, HCI research on method assessment and comparison still largely proceeds on the basis that methods can be adequately described and reliably applied with some guarantee of outcome. This ignores both the limits of methods within design and evaluation practices and the realities of usability work, which we both now briefly review.

2.1. Methods, as Loose Collections of Resources, Are Weak Prescriptions

The word *method* can seduce the unwary into thinking that human behavior (in the case of Interaction Design, highly skilled creative knowledge intensive work) can be extensively constrained and programmed. Developers who submit to a method's discipline are thus guaranteed results. Thus we could be lead into thinking that we could objectively compare methods that are correctly applied. However, this is misconceived, because (in)correct application does not ensure (failure or) success. Although it is clear that methods can be incorrectly applied, this does not automatically lead to poor outcomes. For example, successful predictions of usability problems with HE can be associated with inappropriate heuristics (Cockton & Woolrych, 2001). A poor choice of heuristic will not necessarily stop problems being incorrectly predicted.

Methods are ideals that supposedly prescribe their usage, perhaps not fully, but at least to a sufficient degree that users are not left to make critical or challenging decisions without support or guidance. Methods create the expectation of a prescribed series of steps or stages, with well-defined decision points, and extensive guidance on what to do at each point. An ideal method guides choices. In reality, however, no method can fully prescribe its usage. To introduce a culinary analogy: Recipes do not cook themselves, and recipes must omit many crucial details. And following a recipe, as all amateur cooks are aware, does not guarantee success, nor does it answer all the cook's questions. No two cooks will produce identical dishes from the same recipe; even the same cook may fail to do so consistently.

What, then, do HCI methods ask of their users? Consider user testing, for example. Although often spoken of as a "method," it is really a broad *approach* that provides little guidance beyond requiring someone to use an interactive artifact and recording the outcomes. The "user testing method" does not tell usability practitioners what to test, with whom, or how. It does not tell them when to test, where to test, or even why to test. Practitioners are thus left with many project specific decisions about, for example, participant recruitment and selection, test task/activity design, measures and data collection, briefing and debriefing, data analysis, and results presentation. This is hardly a "recipe." Even in those practitioner texts that are clearly detailed supportive "cookbooks" (e.g., Courage & Baxter, 2005; Holtzblatt, Wendell, & Wood, 2005; Mayhew, 1999), there is a wide variation in detail on key usability activities such as recruitment and selection of participants for user testing and similar activities. Where both broad and detailed systematic, step-by-step guidance is given (e.g., the 34 pages of advice on participant recruitment in Courage & Baxter, 2005), the examples and practices reflect their origins (in this case, large U.S. software development corporations). The advice assumes similar origins and fails to cover, for example, situations where participants are recruited by an external client for usability work. Even where the organizational contexts match, usability specialists and/or their collaborators must develop their own screening questionnaires. Although Courage and Baxter's (2005) two-page detailed example screener is helpful, it is only an example that must not be adopted mechanically for new situations.

User testing is a somewhat extreme example of an evaluation "method," strictly covering a family of approaches that record and measure user interactions with (prototypes of) digital artifacts. Even so, most HCI design and evaluation "methods" ask for at least as much knowledge, initiative, and resourcefulness from their users as cooking recipes do, and often much more. For example, Courage and Baxter (2005) advised that product teams must help you to develop participant recruitment criteria but gave no example procedures for doing so. Similarly, HE requires evaluators to choose from a range of possible discovery procedures, such as system scanning, system searching, goal playing, and task method following (Cockton, Woolrych, Hall, & Hindmarch, 2003). Unlike a recipe, HE does not even offer choices of alternative "ingredients" here, with only occasional mentions of ways to carry out a usability inspection (Cockton, Lavery, & Woolrych, 2008).

As a further example that methods as published do not determine usability outcomes, Jeffries, Miller, Wharton, and Uyeda (1991, p. 122) asked evaluators to "report problems they found even if the technique being used did not lead them to the problem, and to note how they found the problem." They classified evaluators' answers as to whether the problem was found "via technique," as "side effect" (e.g., when applying a guideline about screen layout, problem with menu organization might be noted), or "from prior experience with the system." Jeffries et al. showed that evaluators using guidelines report many problems found as side effects (23%) and from prior experience (34%). Hornbæk (2010, p. 102) considered this an instance of a dogma in method assessments that "Usability Evaluation Proceeds as Prescribed and Directly Identifies Problems." Research on HE has systematically exposed the need for skilled use of resources that HE itself cannot provide (Cockton & Woolrych, 2001; Cockton et al., 2003; Cockton, Woolrych, & Hindmarch, 2004; Woolrych, Cockton, & Hindmarch, 2005). These insights have been synthesized within a model of usability work that attributes success to the effective combination of problem *D*iscovery and *A*nalysis *Re*sources (DARe model; Cockton et al., 2008). The refocusing of HCI research advocated in this article has its origins in the DARe model, but overall this article's position is based on similar discoveries across a wide range of recent usability research.

All methods ask something of their users, but some ask more than others. As the skill and knowledge requirements increase, irrespective of the level of detail in cookbook publications, methods become less prescriptive, sometimes to the point of near invisibility, as in user testing. All HCI methods require extensive designer and/or evaluator configuration. There are no off-the-shelf methods that can be faithfully followed "as is." Methods are collections of resources that must be configured/complemented for use in specific project contexts.

The position that methods are weak prescriptions should not surprise the HCI community. Although the evidence supporting the DARe model is relatively new, the fact that plans are resources for situated actions is not. More than two decades have passed since Suchman's (1987) landmark critique of intelligent photocopiers. Suchman's research readily exposed how the plan recognition built into an intelligent photocopier was no match for the variety of real user behaviors. By treating a plan as a script, the intelligent photocopier was too rigid and unimaginative in its interpretation of user behavior. Suchman argued that plans should be regarded as resources that have some fit to potential real-world situations, but, to achieve fit, such plans need to be modified and extended to cope with the situated realities of human behavior. If we accept that an intelligent appliance is highly unlikely to ever adequately anticipate or plan for the situated nature of human behavior, then it should be clear that authors of usability methods face an even bigger challenge, because usability work will generally be far more challenging and complex than, for example, photocopying. We need to remove such inconsistencies within HCI thinking. If we cannot script user interaction, then we cannot script usability work. Usability approaches must then be designed, understood, and assessed within a framework that recognizes the situated nature of all human activities. Within such a framework, evaluation methods can never be anything more than weak prescriptions. We thus need to drop our focus below the level of methods to the underlying resources that are configured and combined during usability work.

2.2. Usability Work Mixes and Fixes Methods

Much method research in HCI is focused on supporting the choice of *a* method, attempting to answer questions such as what, in some specified context, is the "best" method, where "best" may mean most productive, most thorough, most valid, easiest to use, or cheapest to use, casting usability work as a simple choice of methods. Given a problem in usability work practice, there should be a method that can best solve it. Real-world choices, however, can never be specified wholly in terms of options and selections. Thus Furniss (2008) argued that an appropriate choice and use of a method is functionally coupled to the context, including client biases, practitioner expertise, their relationship, the budget, the problem, the time, auditing potential (for safety critical systems development), and persuasiveness.

Although HCI research that compares evaluation methods can be based on an unrealistic position on the power of methods, usability practitioners cannot base their work on such illusions. Extensive project-specific requirements (e.g., choosing participants, reporting results) distance methods in use from published methods, even when these superficially appear to be detailed, step-by-step practitioners' "cookbooks." Thus, usability work requires a *range* of resources to be combined (Rosenbaum, 2008), generally compensating for each other's weaknesses, reducing or even removing the relevance of rankings or assessments of isolated methods. Molich, Ede, Kaasgaard, and Karyukin (2004) showed how teams that tested the same website differed on many specific choices on how to conduct a usability test. For instance, with respect to test reporting and task selection, marked differences were found. The nine teams not surprisingly found different sets of problems. Surprisingly, very few scientific studies of methods look at the combination of methods (though see Uldall-Espersen, Frøkjær, & Hornbæk 2008); this supports our argument that very few comparative research studies investigate evaluation methods as they are used in practice. For example, Rosenbaum's (2008) account of the evolution of usability practice shows a clear move over the last decade to project-specific combinations of multiple methods. Within our culinary analogy, usability work corresponds to *cooking a meal*, not an individual *dish*.

2.3. HCI Methods Assessment Research Is Still Overambitious

HCI researchers have repeatedly attempted to assess methods, often by comparing the performance of different design methods or evaluation methods. Given that all methods must be configured for project-specific use, controlled studies must provide such a configuration too, but this may not be representative of how methods are appropriated within practical usability work. What is assessed or compared is thus a specific instance of a method in use, often artificial use. For example, participating evaluators may not be given the time or support to familiarize themselves with the test product and its design/business goals. Also, they may be given only an hour or two to carry out an evaluation that could take at least a day in typical practice. Reliable generalizations from such artificial situations are hard.

A further problem arises with making judgments on whether methods "work" or explaining how they work based on formal studies. It is the configured and contextualized method that does or does not work, that is, *a* method in (artificial) use rather than *the* method on paper, with clear implications for the generalizability of explanations as to why methods do or do not work, and any associated implications for best practice. It is no surprise that such research is seen to fail practitioners (Wixon, 2003).

The disconnect between research on methods and practical usability work goes beyond isolated issues such as problem counting as an assessment criterion (Wixon, 2003). Artificial use of methods in isolation cannot provide reliable insights into use in combination. Also, even if we could compare methods, we would have far too many methods to compare to be able, as researchers, to provide the insights that practitioners need. Thus it appears that we need a different focus on usability evaluation methods. All the authors of this article have published comparisons of evaluation methods, either by directly comparing the performance of two or more methods or by using one method (e.g., user testing) to scope the capabilities of another (e.g., HE). The thinking behind this article has its origins in firsthand experiences of the difficulties involved in making rigorous reliable comparisons at the level of indivisible wholes for evaluation methods. Also, when explaining the different performances of evaluation methods, we increasingly resorted to isolating their "active ingredients," and thus realized that looking *inside* evaluation methods was of much more relevance and value that looking *at* them.

3. TOWARD INGREDIENTS AND MEALS

An alternative strategy to focusing on methods is to study their component resources, such as problem reporting formats, participant selection strategies, and design heuristics. Figure 1 illustrates the move from methods to resources. In our culinary analogy, we argue that we need to drop *below the level of recipes to ingredients*, where recipes are methods as published. Ingredients, then, are the resources that make up published methods.

The antecedents of practice are incomplete resources, not complete methods. Thus, research should focus on how resources are configured, combined, and contextualized. It is also the case that evaluation method designers' influence is necessarily greatly limited by project specific configuration, contextualization, and combination of methods. We need to also go *above the level of recipes and ingredients to cooking a meal* (Figure 1). When cooking a meal, a recipe may be required for each dish, but cooks also need to source ingredients and provide appliances such as stoves, freezers, and fridges. Similarly, methods need to be studied in genuine project contexts, with a focus on the interaction between the resources provided by methods, those (not) provided by evaluators, and those (not) provided by the project or organizational context.

Illustrating the feasibility and benefits of these methodological shifts is the main concern of the rest of this article. Feasibility is demonstrated by identifying the resources within three well-known HCI evaluation methods. This is reinforced by

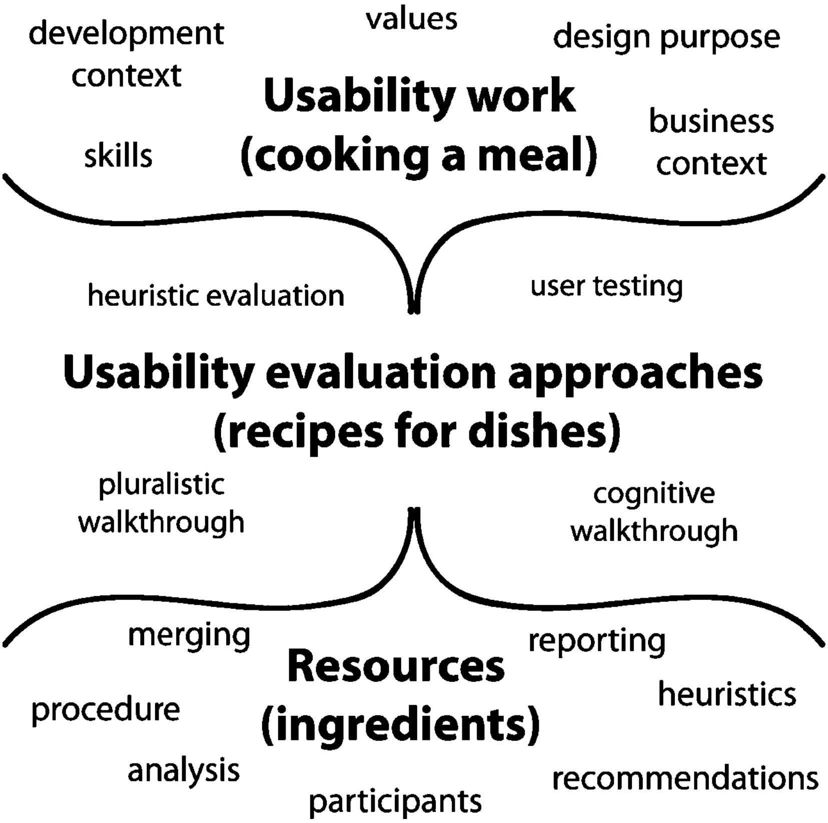

FIGURE 1 We argue the need to go from methods to resources, and from methods as prescriptions to usability work.

identifying resources that have already been a focus of HCI research, both in the context of a specific method and as auxiliary resources that can be used with a range of methods. Together, this evidence from existing research shows that we can respond to the need to drop *below the level of recipes to ingredients.*

To go *above the level of recipes and ingredients to cooking a meal*, we need to understand how usability work configures and combines methods. By identifying the resources within existing methods, we can identify implicit needs for complementary resources that must be provided by evaluators and/or the project context. To address the latter, we briefly review research on information systems (IS) methodologies and studies of usability work at the project and/or organizational level. This complements the bottom-up analysis of the next two sections with a top-down analysis that scopes out a wider potential space for evaluation resources.

These brief reviews (resources within methods; resources complementary to methods; macroissues of methodology; projects and organizations) form the basis for a closing discussion of the implications of our two advocated methodological shifts for research, practice, and teaching.

For the remainder of the article, we use the term *approaches* in preference to *method*, to avoid the misconceptions associated with the latter. Approaches are understood to be people's usage of loose and incomplete collections of resources. Some of these resources can be used passively without adaptation (e.g., lists of heuristics), whereas others must be configured for specific projects (e.g., task specification notations). Practitioner texts thus stress the need for flexibility, making it clear that if, for example, a carefully planned usability test protocol is not delivering, it should be revised on the fly.

4. EXAMPLE RESOURCES WITHIN EVALUATION METHODS

Table 1 summarizes evaluation resources that are frequently discussed in HCI research. There is not space to describe or motivate them all here (several are covered in more detail next). The analysis in Woolrych et al. (2005) provides more examples of resources in usability inspection. A recently completed European project (Cockton & Woolrych, 2009) has also published a report on resources and their relation to evaluation method comparison. To illustrate how the resources within evaluation approaches may be identified and reasoned about, we first focus on resources provided by HE, cognitive walkthrough (CW), and user testing. As a

Table 1: Examples of Resources and Research Investigating Variants of Those Resources

Resources for	*Focus*
Participant recruitment	Finding the right kind and number of participants (Gilbert, Williams, & Seal, 2007).
Task selection	Selecting and specifying tasks for inspection or user testing (Lindgaard & Chattratichart, 2007; Sears & Hess, 1999).
Problem merging	Identifying similar problems, for instance through tool support (Howarth, et al., 2007), or using different definitions of similarity (Connell & Hammond, 1999; Hornbæk & Frøkjær, 2008a).
Task walkthrough	Supporting inspection methods with ways of going through tasks and interfaces (Sears, 1997).
Reporting formats	Helping communicating problems and solutions for subsequent analysis, evaluation auditing, iteration, and customer communication (e.g., Capra & Smith-Jackson, 2005; Theofanos & Quesenbery, 2005).
Problem identification	Tools and approaches for identifying/discovering problems from empirical test data (Skov & Stage, 2005).
Problem classification	Ways of classifying problems so as to analyze, merge, or reject them (Andre, Hartson, Belz, & McCreary, 2001; Cuomo & Bowen, 1992).
Analysis	Approaches to synthesizing and understanding data from usability evaluations, e.g., Instant Data Analysis (Kjeldskov, Skov, & Stage, 2004), DCART (Howarth et al., 2007), SUPEX (Cockton & Lavery, 1999), DEVAN (Vermeeren, 2009).
Think-aloud protocols	Variations in how users are instructed to think aloud, such as classic or relaxed (Hertzum et al., 2009).
Heuristics	Resources for discovering and thinking about usability defects (e.g., Chattratichart & Lindgaard, 2008; Frøkjær & Hornbæk, 2008; Hvannberg et al., 2007; Mankoff et al., 2003; Somervell & McCrickard, 2005).

further illustration, in the next section we review three complementary resources that are common to a range of evaluation practices.

4.1. Decomposing HE

HE in its original form (Nielsen & Molich, 1990) provides three resources: 10 heuristics, the reasoning behind them, and an outline procedure for applying them. Nielsen (1994) subsequently refined this procedure by recommending that analysts go through the interface at least twice. The first "pass" allows the evaluator to get a "feel" for the system, that is, both the general scope of the system and the flow of interaction. This prompts evaluators to add a fourth project-specific resource—knowledge of the system under evaluation. In the second and subsequent passes, the evaluator can focus on specific interface elements that breach one or more heuristics (Nielsen, 1994). The loose coupling between approaches and some of their resources is evidenced by Nielsen revising the recommended procedure.

These three resources are not sufficient to explain the performance of evaluators using HE. Cockton and Woolrych (2001) demonstrated that many outcomes of HE could not be attributed to any of these resources, replicating earlier research by Jeffries and colleagues (1991). Cockton and Woolrych used focused user testing to confirm that evaluators had successfully predicted some usability problems with which no heuristic could be credibly associated. Generally, heuristics were often inappropriately applied (poor association of heuristic with problem), being most appropriately associated with problems that turned out to be of low frequency and/or severity in user testing. HE's resources could not have been responsible for many of the one third of predictions that were confirmed by carefully focused user testing. The effective resources here had to come from the evaluation context, that is, from the student evaluators themselves who were provided with a lecture and training manual that covered HE's resources. Missed problems were due to deficiencies in the combined resources of both HE and the evaluation context, which failed to compensate for HE's weak problem discovery resources. In particular, Nielsen's recommended first scan often did not equip evaluators with adequate knowledge of the system being evaluated, leading to both missed problems and dozens of bogus ones that were factually incorrect about the system's capabilities. The inadequacies of this "two-pass" procedure and the heuristics themselves were further suggested by the user testing, which also revealed serious problems that had been missed by 20 groups of almost 100 student evaluators.

The benefit of adding new resources to HE was shown in two follow-ups to Cockton and Woolrych (2001), where an extended structured problem report format was shown to have a clear replicated impact on evaluator performance, almost doubling the appropriate application of heuristics and increasing validity from 31% to between 48% and 63% (Cockton et al., 2003; Cockton et al., 2004). Note that these significant improvements in evaluator performance were not due to changes to the inspection method. Instead, a resource external to HE, that is, a mostly generic problem report format, was the source of the improvements. The extended structured problem report format prompted both discovery resources (often improving on Nielsen's first pass) and analysis resources (through asking for reasons for keeping or eliminating a potential problem). Nielsen has since recommended that evaluators give reasons for reporting usability problems (Nielsen, n.d.).

Subsequent analysis (Woolrych et al., 2005) of the data from these studies (Cockton et al., 2003; Cockton et al., 2004) identified seven categories of *distributed cognitive resource* (DCR), so called because these were distributed across the context of an evaluation rather than being provided by any approach being used. DCRs were shown to have positive impact on successful predictions using HE (true positives and negatives) and a negative impact on unsuccessful predictions using HE (false positives and negatives). This demonstrated the impact of resources that were not provided by the HE approach, with a clear evaluator effect in terms of which complementary resources were used effectively and when. However, evaluator effects were not consistent, with successful use of complementary resources for some predictions but poor use for others. Currently seven DCR categories have been identified in studies of HE usage:

1. knowledge of users and their abilities,
2. task knowledge,
3. (application) domain knowledge,
4. knowledge of the tested product/application/system,
5. knowledge of the implementation platform,
6. knowledge of how users interact with computers, and
7. knowledge of interaction design options and their consequences.

In all three evaluation contexts, the evaluators had to provide these DCRs themselves, except for two "approach-free" evaluators in Cockton and Woolrych (2001) who were prompted to use a basic domain resource (diagrams to be copied from a textbook) as a source of user goals, which demonstrably improved their problem discovery.

Few required DCRs are present as resources within HE. The heuristics in HE communicate some knowledge of interaction design (the seventh DCR item just listed), but the first "feel" pass results in inadequate product knowledge (fourth DCR item). All other DCRs must be sourced within the evaluation context when applying HE, either as part of the knowledge and skills of evaluators or as information provided by other roles in the development team. The quality of evaluator performance appears to be heavily dependent on the availability and successful application of these DCRs. The absence of DCRs, or their misapplication, can result in distinctive defects in evaluation performance. Misapplication of DCRs can be responsible for a significant proportion (if not all) of "false positives," particularly flawed resources associated with perceived user knowledge (Woolrych et al., 2005). Specifically, an underestimation of user abilities results in predictions of usability problems that are often so easily overcome that no actual test users were ever affected.

4.2. Decomposing CW

CW is the best known inspection technique motivated by theory (CE+; see next). In the originators' final version of CW (Wharton, Rieman, Lewis, & Polson, 1994), this approach's resources were reduced to four questions and a procedure for stepping through a task description. CW provides no resources for task selection or support the construction of CW's *success* and *failure* cases. Early versions of CW prompted evaluators to gather a wide range of background information corresponding to several DCR categories (Woolrych et al., 2005) such as user and domain knowledge, and product features such as input device usage. This prompting creates similar inconsistent resources as Nielsen's first HE pass previously. These prompts were removed due to problems in teaching CW in time-constrained classroom settings (Cockton et al., 2008). Given that the required resources for user, domain, and device knowledge are distributed in real evaluation contexts, a classroom is a poor surrogate. It is no surprise that CW novices in tutorial settings found it hard to apply such a wide range of background resources. The lack of advice on determining success and failure cases is more puzzling, because these form CW's critical decision points. Unsurprisingly CW is associated with (very) low rates of problem discovery (Cockton et al., 2008).

The backgrounding of the CE+ theory of learning within later versions of CW (Cockton et al., 2008) is also interesting, suggesting that some resources fare better than others. There appears to be a tendency to favor the concrete (e.g., notations, heuristics) over the abstract (theories, concepts), and the practical, (e.g., procedures) over the propositional (e.g., knowledge, information). Here, short-term learner preferences could work against long-term effectiveness of approaches by devaluing the propositional resources that actually "make them work." The reality is that practitioners *have to work at* methods. Any expectation of rapid true competence is a dangerously misleading expectation.

Spencer (2000) reported a further simplification of CW. In what he called the streamlined cognitive walkthrough, only two questions are asked, neither with reference to psychological theory: "Will the user know what to do at this step?" and "If the user does the right thing, will they know that they did the right thing, and are making progress towards their goal?" (p. 355). Of interest, Spencer added several procedural resources to the method, including rules about not discussing cognitive theory and not designing during a walkthrough. These resources appeared to work well in a large-scale, commercial development context.

An evaluation procedure resource is a key part of CW. It prompts evaluators to carry out a task-based analysis of users' ability to plan, find, understand, and interpret. Although analytic evaluation methods may vary on many dimensions, the procedure for conducting the evaluation appears to strongly determine performance. Early studies showed that evaluators found CW's procedure tedious: John and Packer (1995) found that going through a CW was "very tedious," and Rieman et al. (1991) showed that computer support for CW increased evaluation efficiency markedly. Despite these attempts to work on CW's procedural resources, only a few studies of which we are aware have evaluated variations of evaluation procedures. There are many open questions on procedural resources, such as the relative impact of self-guided exploration (Nielsen, 1994) versus structured walkthrough on inspection performance. Structure has been shown to be as effective even when a system-centered rather than a user-centered approach is taken (Cockton et al., 2003). Despite being tedious, the resource of structuring a walkthrough can strengthen other evaluation approaches that lack it. For example, Sears (1997) showed how combining CW's strict task walkthrough procedure with the loose sweep of HE was more effective than HE alone. Analysis of HE heuristics indicates that several cannot be assessed without recourse to task analysis (Lavery, Cockton, & Atkinson, 1996).

4.3. Decomposing User Testing

Whereas user testing appears to be a simple case of a method, it is not. No documented description of user testing tells usability practitioners what to test, with whom, how, when to test, where to test, or why to test. Most of these questions are too project specific, not only to the digital product or service under development but also to stages of development at which these are evaluated. The best usability practitioners can hope for is generic advice, detailed tutorial examples, and example case studies that cover specific resources such as participant recruitment and selection, test task/activity design, evaluation location and setups, measures and data collection instruments, briefing and debriefing procedures, data analysis, and results presentation. Each of these resources is complex, requiring more space to cover than is generally given to presenting mature "methods" in the literature. Practitioner texts rarely devote more than a few pages to participant recruitment and screening. Even when 35 pages of advice on recruitment and selection are provided (Courage & Baxter, 2005), these do not extend beyond the authors' experiences.

Attempts to compare user testing to other evaluation methods are understandably fraught methodologically (Gray & Salzman, 1998), because a single resource can bias results. For example, Hertzum, Hansen, and Andersen (2009) showed how changing one of the basic resources in think-aloud testing, the instruction to participants of how to think aloud, affected the evaluation process and outcome. The relaxed way of thinking aloud, used in most usability tests in practice, resulted in higher workload, longer task completion times, and different behavior on the website under test. Similarly, Nørgaard and Hornbæk (2006) discussed how practitioners who analyzed think-aloud tests relied on ad hoc and unsystematic approaches. Here, attention to variants of a key resource provided important insights.

Although many aspects of think aloud have been extensively described, few comparisons of approaches report how evaluators proceeded from test data to problems and to ideas for solutions. Cockton and Lavery (1999) discussed approaches to problem extraction from empirical data but provided only a framework (SUPEX) for analysis, not a stepwise procedure. Vermeeren and colleagues have developed the DeVAN method and variants that are more concrete than SUPEX but still require project specific decisions to be made during use to extend DeVAN into a complete method (Vermeeren, 2009). SUPEX and DeVAN produce different problem sets to unstructured problem extraction and can even be configured to produce different problem sets from the same data, depending on the researchers' interests. Given this, objective comparisons between user testing and other evaluation approaches can never be straightforward.

Some relevant resources are not unique or specific to user testing but are instead transferable between methods. When comparing, for example, user testing and CW, careful consideration is needed of their overlapping resources. Differences in outcome could be due not to the overall approach but to specific differences in configuration of a common resource. For example, CW evaluators provided with typical user profiles could find problems that user testers may not, because CW could cover tasks where typical users would not have the knowledge needed for successful task completion, but user testing would not. Although one bad apple does not spoil the bunch, in comparisons of usability approaches, one uncontrolled resource often spoils a comparison. The devil lies in the details, and the fortunes of evaluation approaches depend absolutely on the details. Whenever one approach has been shown to perform better than others, knowledge of evaluation resources makes it possible to imagine plausible circumstances in which this would be reversed.

User testing thus relies on specific resources, such as think-aloud, problem extraction, or participant recruitment and selection techniques, and transferable resources such as user profiles and task descriptions. For some of these, practitioner texts provide detailed examples and cookbook-style instructions, but these are only a partial answer to providing generic project-independent resources. As a complement, a deep understanding of relevant principles, and the ways in which they have been effectively applied in a range of project contexts, could have more value and better impact than a set of examples and case studies from a single source. Although we prefer to learn from concrete examples, deep understanding depends on a strong grasp of abstract principles, for example, concepts of experimental bias, forms of priming, and informed consent.

5. RESOURCES COMPLEMENTARY TO SEVERAL EVALUATION APPROACHES

The previous analyses of HE, CW, and user testing demonstrate that it is possible to look "inside" approaches and identify the main component resources; Table 1 summarizes some of these. However, other evaluation resources have already been researched independently of specific approaches. Such resources can complement or replace resources in established evaluation approaches. Existing research here increases our confidence that we can profitably move the focus in HCI evaluation research from monolithic methods to component resources.

5.1. Resources for Selection and Specification of Tasks

Test task/activity design and specification are a core resource for many inspection and model-based approaches, as well as for fixed task user testing, and for data-mining-based approaches for automatic analysis of usage logs. Second only to individual differences between test participants, the selection of particular tasks has a large effect on interaction with computers, and hence on evaluation outcomes. Lindgaard and Chattratichart (2007) provided evidence that the selection of tasks increased the problem yield more than the number of users in a test did. This extends earlier findings (Woolrych & Cockton, 2001) where a change in evaluation tasks resulted in new problems being found. Sears and Hess (1999) described how the amount of detail in task description affected which problems were found in CW. Yet almost no studies have empirically investigated the impact of various ways of selecting tasks on evaluation results. For instance, there is an obvious difference between user-generated tasks and set tasks, and between open-ended and closed tasks (Spool & Shroeder, 2001). We consider this one promising underexplored area for investigating the impact of a single resource. There is initial research to build on, but many important research questions have yet to be systematically addressed. For example, the impact of notational resources for task specification, such as NGOMSL (Kieras, 1988), has not been systematically investigated.

5.2. Resources for Usability Problem Merging and Matching

In virtually all usability work, individual problems that are instances of more general and/or frequent user difficulties are predicted or discovered. Such individual problem instances need to be matched to reduce a large set of instances of specific predictions or user difficulties to a master set of problems grouped by type, with associated frequency data, severity ranges, and detailed instances.

Problem matching and merging is a major source of confounds when comparing usability methods (Cockton & Lavery, 1999; Connell & Hammond, 1999; Hornbæk & Frøkjær, 2008a; Lavery & Cockton, 1997; Lavery, Cockton, & Atkinson, 1997). This is not only a problem for research: Matching and merging are very rarely avoidable in usability work, and thus evaluation resources are needed to provide matching constructs, merging criteria, and reporting formats. Examples of all are currently available in the literature, but we know of no formal comparison studies other than Hornbæk and Frøkjær (2008a). Once again, there is extensive scope for research focused on resources below the level of evaluation approaches. By focusing on specific uses of resource variations, the major methodological obstacles associated with comparing methods can be significantly reduced.

5.3. Resources for Problem Reporting

Problem report formats not only are a critical resource for matching and merging for researchers and practitioners but also have been shown to improve evaluator performance when combined with HE. The extra effort required to complete complex formats increases the validity (Cockton et al., 2004) but not the thoroughness of a method (Cockton et al., 2008). However, there are alternative resources to report formats that do not add to the cost of reporting every problem instance. For example, group procedures for assigning problem severity and fix prioritization can do much of the work of problem formats, matching, and merging resources through discarding invalid problems in passing during discussions.

Only a relatively few studies have investigated the influence of how problems are reported on the outcomes of evaluation and how such formats influence the evaluation process. The results of Cockton et al. (2003) have already been noted, with an extended report format positively impacting evaluator performance on false positives and appropriate HE use. It has also been suggested that reporting justifications for why something is a problem is important (Jeffries, 1994) and that the downstream utility of problem descriptions seems to increase with such justifications (Hornbæk & Frøkjær, 2006).

One example of work that enhances problem reporting resources for a variety of evaluations focused on business goals (Hornbæk & Frøkjær, 2008b). Evaluators in a think-aloud test were instructed to consider business goals while evaluating and to report, together with typical parts of problem reports, the importance of a problem in relation to business goals. Compared to a control group, this format made evaluators report fewer problems, but the problems reported were seen as being of more utility in the development process. Of interest, this resource could be used with many other evaluation approaches and might be particularly useful when evaluators have limited knowledge of the business goals behind an application. It is a small change with a potential large impact.

One problem report format is supported by a tool, DCART, that supports usability practitioners in going from usability data to usability reports (Howarth, Andre, & Hartson, 2007), including identifying problems and merging them. Problems found in the DCART use condition were rated significantly higher by developers (on quality) and independent raters (on usefulness). However, explaining the success of DCART is not straightforward, because it supports several procedures in addition to merging, in particular the automatic creation of usability reports: Impact cannot be ascribed with confidence to specific DCART resources such as the process support tool. Nevertheless, these procedures are resources in the sense that they may be combined within other evaluation approaches; the authors behind DCART (Howarth et al., 2007) suggested that they may be particularly useful for novice evaluators.

Problem report formats and practices as complementary evaluation resources have been studied more than other generic resources. Even so, there are many unexplored questions relating to research and practice. Along with task specification, task selection, and matching and merging resources, problem report formats are an example of resources below the method level that can support a wide range of evaluation approaches, and can thus provide a profitable focus for HCI research.

6. TWO COMPREHENSIVE PERSPECTIVES ON EVALUATION RESOURCES

We have identified a range of evaluation resources in Table 1, which was complemented by detailed analyses of HE, CW, and user testing, as well as more general resources (task specification notations, task selection procedures, matching criteria, and merging procedures, problem report formats). Together, these are evidence that research can focus within approaches on component resources. However, the resulting set of identified resources feels arbitrary, and we thus need to consider whether more comprehensive views of evaluation resources are possible.

In this section, we move from the bottom-up analyses of the previous two sections to two top-down analyses. The first draws on established theory from IS methodologies to support a systematic analysis of abstract resource types. The second briefly reviews research on usability work that has studied the wider context of method formation and use, identifying project-specific resources that can have a stronger influence on evaluation success than the resources provided by generic (i.e., project-independent) evaluation approaches.

6.1. The Scope of Resource Categories

To complement our somewhat random bottom-up sets of identified evaluation resources, we need to look for more comprehensive top-down structures. In the 1980s, IS researchers were engaged in reflective grounded analyses of methods and methodologies. This is a basis for deriving an initial broad abstract view of resources for design and evaluation. Three decades ago, IS consensus built around a conceptualization of method that was well expressed by Mathiasen (1982), which, in brief, considered the concept of a system development method to comprise the following components:

1. An application *area*, an application *domain*
2. A *perspective* on the software development processes and the goal to be achieved
3. *Principles* of organizing the software development processes
4. *Tools* (including notations, description forms, instrumentation, automation)
5. *Techniques* (i.e., work processes for people in the software development processes)

From an HCI perspective, within tools (the fourth component), we would want to distinguish specific (the fourth component) tools specific *evaluation* resources such as *instrumentation* from *design* resources such as *notations*. Also, given the role of knowledge and theories from the human sciences in HCI research and practice, we should also add a sixth *knowledge* component. Translating the resulting list into broad classes of resource, we have the following:

1. *Scoping* resources that indicate the extent of a method's applicability in terms of the purposes and usage contexts of what is being designed or evaluated, including application areas/domains
2. *Axiological* resources that indicate the values underpinning a method (perspectives)
3. *Project management (process)* resources that situate a method within an embracing development and collaboration context
4. *Expressive* resources that communicate the output of a method via specifications, reports, and so on

5. *Procedural* resources that guide use of a method, including partial automation through tools
6. *Instrumentation* resources that collect issues and measures for evaluations
7. *Knowledge* resources that underpin one or more of the previous resource classes

IS and software engineering research on methodology in the 1980s was more ambitious than it is now, with grand structures giving way to flexible agile approaches to software development. Despite this retreat from an apparent attempt to understand and manage everything, software engineering and IS research from the 1980s does prompt the question as to whether HCI research on development methodology should replace waning ambitions on validating complete methods with ones on understanding the utility of a broad range of design and evaluation resources.

Although abstract, this list of resource classes has immediate value, as it exposes an imbalance in the range of evaluation resources that have been developed and assessed within HCI research. Method research in HCI has tended to focus on the last four broad classes just listed, ignoring the specificity of the first and the organizing constructs of the second and third.

Consider axiological perspectives in the aforementioned conception of method: a *perspective* as a resource frames practitioner expectations. A perspective is not the ultimate answer to all problems in usability practice, but it does provide vital cohesion for an approach. For example, CW's perspective is to smooth or harmonize the relation between task execution and visible user interface elements. Similarly, the Metaphors of Human Thinking approach (Frøkjær & Hornbæk, 2008) has the perspective of harmonizing interaction with habitual mental and physical activity, as well as some of the limitations that we share as human beings (difficulties in maintaining concentration, misunderstandings, mental slips, etc.).

Relations between broad classes of resource preconfigure work systems that structure and shape the combination, appropriation, and use of specific resources in practice. For example, expressive resources need to have strong synergies with the core concepts of an approach's philosophy. Also, consistency and synergies are needed between an approach's *axiology* (what it values and holds worthwhile) and knowledge and procedural resources. Method design and evaluation has tended to emphasize procedural and knowledge resources at the expense of expressive and axiological resources. Procedures are only one form of resource within approaches, and they may hardly be present as a detailed resource (as in HE, and even from some perspectives in CW). Given that the existence or extent of detail for procedural resources can vary substantially, method innovation and evaluation should give more attention to nonprocedural resources. Research needs to focus more on knowledge, perspectives, scopes, and expressive resources.

6.2. Factors Influencing Usability Work

We make no attempt here to survey all of the factors that have been shown to shape usability work, but instead we focus on some research that considered the broader context of method use. Theofanos and Quesenbery (2005), for instance, listed a variety of factors that shape how to report results, including the size of the recipient company, the kind of products evaluated, the audience, and the existence/absence of a formal usability process. Similarly, the software *development approach* used in a development project influences usability work. Sy (2007) described how various approaches to usability evaluation had to be modified to fit the agile software development approach. For instance, reporting of problems was done through the daily scrum meetings and in process planning sessions, rather than through reports, and thus specific evaluation resources were not needed here (although usability roles may well have kept their own informal records).

Here we briefly list several factors of which we are aware from research that has addressed how project- and organization-specific factors can influence the configuration and combination of evaluation resources:

1. Client needs and expectations
2. Design purpose and vision
3. Project, product, and service specific prioritization criteria
4. Business context
5. Budgetary and other logistical resources
6. Project leadership
7. Development approach and stage
8. Relation of usability work to the overall software development approach
9. Experience and competence of usability practitioners
10. Training and tutorial support on evaluation approaches
11. Professional/specialist education on general discovery and analysis resources in evaluation
12. Field research methods (users, tasks, etc.) and result communication formats and tools
13. Task specification and notations used at design stage
14. Participant and evaluator recruitment strategies
15. Alignment of design purpose and evaluation purpose

The first five factors relate to the client and their economic/market context (or policy context for public sector projects). The next six factors relate to the development team, with a specific focus on management and expertise. The last four factors relate to design activities (last two focus on evaluation). These factors, and many others not listed here, severely limit the ecological validity of formal comparisons of evaluation methods, even in the unlikely event of confounds being so well managed that no issues arise there.

Few of the aforementioned factors could be addressed substantially by evaluation resources from HCI approaches. All, however, are present to some degree in actual project and organizational contexts. Currently, the bulk of the resources that are critical to working out methods within usability work are external to documented HCI evaluation approaches. Given this, academic attempts to "compare methods" are bound to "fail the practitioner" (Wixon, 2003), because as far as providing comprehensive support for usability work is concerned, all current documented approaches have already failed the practitioner to some extent by not providing them with comprehensive resources. However, much of the success of "discount methods" such as HE may be due to the sparseness and flexibility of the provided resources, which allows skilled evaluators to fill the gaps with resources from personal, project, or organizational sources.

7. IMPLICATIONS FOR RESEARCH AND PRACTICE

We have argued that a focus on individual evaluation methods in HCI research is conceptually misguided, overestimates method impact, and is out of touch with the realities of usability work. We have proposed an alternative dual strategy of moving both below and above the level of methods: *below methods* to their constituent resources, and *above* them to the contexts of usability work. We have demonstrated the viability and value of such a shift in research strategy by identifying evaluation resources within three existing approaches as well as three resources that have been developed independently of specific approaches. We have complemented this bottom-up survey with a top-down analysis of broad resource classes for information and software systems development, and, last, with a brief overview of factors within usability work contexts that can and do strongly influence the impact of specific evaluation approaches and their combinations.

Through this argument and analyses, we have achieved the primary aim of this article, which is to pull together a range of emergent trends in evaluation research and practice within HCI and interaction design. The authors have been closely involved in these trends and have drawn on their own research for many of the examples in this article. However, there is clear evidence of a wider trend toward a move away from single method comparisons to new foci on evaluation method synergies, on one hand (e.g., Law & Hvannberg, 2002; Sears, 1997), and on the impact of specific evaluation resources, on the other hand (e.g., Hvannberg, Law, & Lárusdóttir, 2007, Law & Hvannberg, 2008).

The value of this article's argument and analyses takes several forms. First, the value of resource combination as a framing device for method research and practice is that it helps us to understand why existing research on HCI design and evaluation methods has struggled to meet the challenge of serving researchers and practitioners. Second, it helps us to recover unappreciated value in existing HCI research by uncovering valuable insights on the impact of resources in studies that were inconclusive on the impact of methods. Third, the analysis supports construction of a road map for future HCI research on evaluation practices. We next discuss these implications for research further, as well as implications for teaching, and practice.

The HCI community also needs to accept that some research goals are infeasible, especially the quest for a single best method; Hornbæk (2010) argued that such a quest is widespread in usability research. The core problem is that, given our argument that methods are a limiting and partly misleading focus of analysis, such a question becomes futile.

7.1. Research Road Map Stage 1: Methodological Studies Should Narrow (to Focus on Resources)

We argue that empirical usability research on methods should focus on evaluation resources. One approach to focusing on resources is to treat them as independent variables in experimental studies. A benefit of this is to avoid practices akin to confounding dishes, recipes, and ingredients; the review by Gray and Salzman (1998) gives examples of such confounding. Earlier sections have given examples of occasional studies that have investigated variations of evaluation resources; we suggest expanding this line of research. Experimental studies on evaluation resources could compare alternatives, for example, different report formats, different data analysis procedures for user test results, different heuristic sets, and alternative think-aloud practices, as well as resources that have yet to be identified and assessed. Such a focus on resources allows more controllable and credible research studies, which would thus have a more realistic chance of providing immediate reliable benefits to practitioners.

Field studies on evaluation resources could focus on the ways in which evaluators configure and adapt resources. Existing resources can be detailed and specific (e.g., heuristics, walkthrough questions) or vague and open (e.g., participant recruitment, briefing and debriefing practices). One area for research here is the impact of detail: How and when do evaluators benefit from detail? CW was subject to a series of simplifications, but it is not clear whether the resulting ease of learning was at the expense of reduced thoroughness, validity, and effectiveness. Similarly, for user testing, evaluators have to choose between broad schema and the detailed step-by-step guidance of practitioner cookbooks. The relative pros and cons of each would be interesting to investigate through research questions such as

- How, when and why evaluators suffer from a lack of guidance on user testing?
- Whether novice evaluators learn superficially about resources from practitioner cookbooks, and copy examples too closely, due to a limited grasp of perspectives, knowledge and scoping resources.
- Where do evaluators make the most mistakes when planning user testing? Where would they benefit from more detailed directive resources, and where can they be left to make their own judgments?

Perhaps the most valuable impact of a focus on specific evaluation resources is that small variations in resources may produce large differences in evaluation outcome. This has already been demonstrated for structured problem report formats, business goals, and so on. Most other resources remain unexplored, for example, the use of product/system information to support usability inspections. Here again, the impact of detail could be explored, and compared to more general strategic resources such as a basic understanding of alternative tactics for structuring and focusing an inspection (e.g., unstructured vs. structured, system- vs. user-focused).

7.2. Research Road Map Stage 2: Empirical Studies Should Widen (to Cover Usability Work in Full)

Another move away from a focus on evaluation methods in HCI research is one that focuses on the factors (contexts and situations) that shape the use of particular resources and their combinations within usability work. Here, the focus is on the impact of specific resources and combinations in specific evaluation settings, that is, how the conditions relating to life-cycle stage, evaluator–designer relation, project size, project visions, known risks, and so forth, impact the course and outcomes of practical evaluation. The focus is thus not on simple choices and combinations of indivisible methods but on the contexts that shape practitioners' appropriation, extension, and combination of resources in usability work.

Although formal experiments are possible when focusing on the impact of specific resources, research that focuses on resource choice and use within usability work requires different research methodologies. To move above the level of evaluation methods in HCI research, we need to make much more use of case studies. Case studies have had some use in research on evaluation methods (Jacobsen & John, 2000; John & Packer, 1995). However, most research to date focuses at the method level, or within them on their ingredient resources.

With reference to the earlier list of 15 factors that are known to influence the outcomes of usability work in practice, recent research and reports on professional practice (Furniss, 2008; Rosenbaum, 2008; Uldall-Espersen et al., 2008) indicate that four contextual factors should be a priority for exploring in future case studies (numbers in brackets refer to the earlier list):

- Experience and competence of usability practitioners (9)
- Design purpose and vision (2), and their alignment to evaluation (15)
- Client needs and expectations (1)
- Problem prioritization criteria (3) (e.g., severity and impact ratings)

The first factor focuses on individual differences. HCI research needs to relax its expectations for scientific objectivity and the goal of eliminating evaluator bias. This is simply impossible except where perfect methods exist that require almost no evaluator judgment. Case studies are required to show how evaluator effects can be *thrilling* rather than *chilling* (Hertzum & Jacobsen, 2001). The best evaluator effects are worthy of imitation; otherwise the worst could endure. However, the quality of an evaluation approach should not be primarily judged (if at all) by its "objectivity," as value judgments are associated with all critical contextual factors (e.g., design purpose and vision, client needs and expectations, problem prioritization criteria). Once evaluation is aligned to such subjective factors, the role of judgment is necessarily increased. Usability work, even when carried out by software generalists, is typically carried out by highly educated individuals who often have postgraduate qualifications. Regardless of qualifications, success in systems development indicates a high level of intelligence. Such individuals are used to exercising independent judgment and expect to adapt methods and approaches to fit local circumstances.

We envisage a research program spanning over a decade, where the demands on research are progressively raised as each challenge is met (Cockton & Woolrych, 2009). A shared research infrastructure of evaluation resources would be valuable for this program, to provide a basis for credible inferences from adequate comprehensive process data. For example, dynamic problem report formats could track the fortunes of usability issues from their identification as possible, through their confirmation as a probable, their acceptance as a priority for the current iteration and their associated design change (or other iterative response), and ultimately to outcomes of these changes during the next set of evaluation activities. Filtering, translating, augmenting, or revising of problem reports may thus provide a basis for understandings of usability work that span all three questions of the RITE method (Wixon, 2003).

The proposed program would begin with detailed, well-structured case studies of usability work. Uldall-Espersen et al. (2008) is a recent example of such a study. Case studies need to focus on key contextual factors such as project leadership and vision. This could involve an action research methodology, allowing the influence of a range of factors to be manipulated as well as explored. Where possible, action research case studies should make careful use of appropriate resources such as structured report formats, not only for predicted, persuasive, and fixed problems but also for the overall process of usability work.

Basing case studies on a common research infrastructure will provide critical support for the second stage of our proposed research program: *metareviews* of case studies of usability work. Individual case studies will always raise issues of generalization, although many concerns here can be offset by the reflective practices of experienced professionals. Anyone acting on guidance from case studies needs adequate critical resources to judge what can reasonably be generalized to their own settings. Even so, the use of a common research infrastructure can and should reduce concerns about excessive subjectivity and a lack of comparability. Indeed, the quality of such comparability will be apparent within metareviews. In time, as more case studies are covered in more metareviews, we should witness a steady increase in the reliability and utility of guidance for usability work.

The third stage of the proposed research program would focus on models of the interaction design process that span complete project life cycles. A key scientific function of metareviews would be to support the *construction of models*, based on credible generalizations over case studies. Such models could provide a basis for well-grounded and socially realistic capability maturity models for usability work. Furniss (2008) created cross-case models of methods in usability practice, which demonstrate important functional components in the context of using “methods” in practice. Furniss developed a *positive resonance model* of how methods fit into a practical context. The model functionally couples methods to a context, with appropriateness and performance dependent on the “push” that they give to a project. Much like a child learns to apply the right push and timing to resonate with a swing, we should understand the resonance that affects a method’s push on a project, considering client, practitioner, and project contingencies. Uldall-Espersen et al. (2008) reported similar empirically grounded models of usability practice.

With models in place, the fourth stage of the research program may finally deliver what was initially sought in formal comparisons of evaluation methods, that is, positive objective knowledge from empirical studies. These could draw on a range of research instruments and procedures, including questionnaires, interviews, and controlled experiments. Positive objective knowledge has remained very elusive in evaluation method research, and will continue to do so if we remain attached to conceptually inadequate experimentation and research instrument design. Current comparisons, investigations, and case studies of evaluation methods do not gather the extent or depth of data from which plausible models can be built, especially ones that can demonstrate the impact of usability evaluation (downstream utility). For credible experiments to finally become possible, we need to pass through several prior stages of case studies, metareviews, and formal modeling.

The aforementioned proposed research program is undoubtedly ambitious, but the reality is that research on evaluation methods in HCI has made do with too little for too long. Furthermore, there are clear trends that leading edge HCI research is already focusing on case studies of usability work (e.g., Furniss, Blandford, & Curzon, 2008) and experimental models of work systems. Elements of the proposed program are thus already in place. However, to get the full value from a corpus of realistic case studies, it is important to follow through with systematic metareviews. Similarly, to validate aspects of complex models, focused studies are required. It is only by working our way as a research community between all four stages of the proposed program that we can finally deliver the quality of research outcomes that are required for high-quality HCI education and professional practice, to which we now turn in our elaboration of the implications of our earlier argument and analyses.

7.3. How to Teach Usability Evaluation

The idea of truth resting on method alone is a very strong one in HCI education settings. It is thus no surprise that students expect evaluation methods to be well-scripted procedures that deliver truth. It is more of a surprise when researchers or educators believe this too, because they will have designed and implemented experimental studies and should fully understand the realities of scientific work. Even so, individual differences in evaluator performance, and contextual differences in the performance of methods, continue to surprise some researchers and practitioners. In contrast, we see these as realities that need to be understood, and where possible adopted or avoided depending on whether their impact is positive or negative.

Realistic teaching on evaluation methods cannot be based on tacit expectations about method in science. Students need to understand the incompleteness of methods and the need for creativity and judgment in designing rigorous experiments and studies. Although the critical understandings here could be an end in themselves, they are also means to other ends, in particular the understanding of axiological perspectives in evaluation practices and the importance of conceptual knowledge resources. Together, these critical perspectives should help to move students beyond superficial learning of notations and procedures. Replacing a focus on supposedly complete methods with one on *approaches*, which gather incomplete sets of resources, should clearly communicate that approaches are not expected to do everything, always require extension (often by combining them), and their resources always need configuration (just as cooks have to peel and prepare vegetables).

The practical realities of usability work are that it is not primarily a question of selecting and combining evaluation methods. In so far as such choices are made, they are the tip of the iceberg. Students need to be familiarized with the seventh eighths of the usability iceberg that lies under the methods surface. It follows that student and practitioner education should focus on resources and their use. Once a useful range of evaluation resources has been covered, existing "branded methods" can be reviewed to determine what they each provide as an approach and what is missing. A solid grounding in resources and the contextual factors that shape their configuration, adaptation, and combination should engender critical abilities for assessing existing and new approaches, and knowing how to combine them. It is such critical abilities that must form the foundation of education on usability evaluation, and not the recalled details of published methods. Given an appropriate grounding in evaluation resources, students could be expected to be able to invent approaches similar to HE, CW, or user testing. The range of resources that they would include would be interesting, especially if they go beyond the skeleton resources provided by existing methods.

Students' understanding of the nature and variable impact of evaluation resources needs to be formed through reading case studies, which can communicate the realities of the complex interaction of configured resources in combination with aspects of development contexts. Research models and overviews can structure this understanding, and the results of sharply focused experiments can guide the design of evaluations. The core understanding, however, is professional: that ultimately practitioners must exercise judgment and take responsibility for evaluation choices. All should understand that there can be deference neither to incomplete decontextualized textbook methods nor to overcontextualized practitioner cookbooks. Students need to understand the broad classes of evaluation resources and how these can interact with contextual factors in specific project or organizational contexts.

Another route is to let novice evaluators begin with learning by doing, by placing them in situations where they can explore the resources that go into usability work, starting with the most important resources in straightforward evaluation scenarios, and gradually introducing complexity in terms of applications, tasks, people, and contexts so as to help novices experience and get a feel for resources and their combinations. Although this is a frequent pedagogical practice in applied arts design disciplines, it is less frequent in science and engineering, where practical work typically focuses on consolidation of taught theories and knowledge. In contrast, practice-led learning begins with direct personal experience and then develops knowledge and understanding from these experiences. Given the complexity and abstract nature of broad classes of evaluation resource, early concrete familiarization with usability practices is important to provide personal bases for understanding theoretical perspectives on usability work.

7.4. How to Carry Out Usability Evaluations

Much of the aforementioned may be no surprise to leading practitioners (e.g., Rosenbaum, 2008). The value to practitioners of the previously mentioned perspectives depends on their professional experience and mindset. For those who still place much trust in following methods, there are major opportunities for personal and professional development in the understanding that methods can be taken apart, reassembled, adapted, and complemented, and yet even then, project and organizational contexts will still be the major determinants of success. By shifting their professional focus within approaches to resources, and above them to working contexts, many practitioners will have a new, professional outlook on their work that will bring opportunities for innovation and improved effectiveness.

Designing evaluations has much in common with designing interactions. Evaluators have to make the best choices they can, implement them, review them, and iterate within the current project or future ones. Cost–benefit assessment is central to usability/user experience. The cost of any completed method is the combined cost of the configured resources. This provides practitioners with a framework for reasoning about cost–benefit trade-offs in usability work. The cost of any additional evaluation resources, or the cost of reconfiguring existing ones, needs to be justified by beneficial outcomes for usability work. Methods do not support realistic cost–benefit analyses here. The argument and analyses just developed takes practitioners both below and above method level, supporting broader reflection on their professional practice. Evaluations will always be constrained by the available budget, so a key aim is to maximize the value that can arise from a specific combination of approaches. This is also why our argument should not be misunderstood as “anything goes.” To abandon method is not to abandon discipline. Rather, it frees up discipline to combine with insight, experience, judgment, and wisdom to reach new heights in the craft of evaluation.

Earlier we mentioned our preference for “approach” over “method.” Practitioners could benefit from this shift in vocabulary, as it communicates existing skills and resources to clients but makes it clear that effective usability work requires careful planning, and clients need to participate actively in this planning. Talking about approaches lets practitioners draw attention to the critical resources that only clients can provide. Even so, we should not forget that the idea of a “method” can be reassuring and can still be used for activities that can genuinely be strongly scripted, such as a style guide conformance review.

8. CONCLUSIONS

HCI should align with the realities of science and design. Both are *work* carried out by individuals and groups, with all the variation and creativity that we should expect when people interpret approaches in specific contexts (after all, this is how HCI does regard users). This is not to argue that there can be no commonalities across design and evaluation practices, nor that best practices cannot exist that offer more success than mediocre ones. However, there will be no absolute consistencies and regularities, and no cast iron guarantees.

The main original contribution of this article is the extensive application of critical analyses to a diagnosis of the ongoing crisis within HCI research on method development and assessment. We have argued that research in support of usability work needs to focus below the level of mythical methods to the actual resources combined within specific approaches. Such research needs to be able to identify the resources within approaches that are combined uniquely and contextually in usability work. Compared to mere method application, such work is like Heraclitus' rivers, never the same the second time you step in them. Just as you can never enter the same river twice, you can't use exactly the same method twice either, at least not without risk of considerable loss of value to software development. Even then, the idea of it being the same method will be an illusion, as choices such as tasks in focus (for testing or inspection), participant briefing, problem prioritization, or problem acceptance criteria must differ from one evaluation to the next, even perhaps when retesting or reinspecting the same application after a remedial design iteration. What are not unique are the resources within approaches and the categories of contextual factors that shape usability. These, and not methods, should be the focus of usability research, teaching, and practice.

The way to resolve this crisis in usability method research is not to abandon it or suppress it, which is what HCI has been doing in practice, but to contribute to a research program that proceeds from realistic case studies, through meta-reviews and modeling to theoretically motivated experimental studies. We have proposed such a research program and related its foundations to the teaching and practice of usability work.

REFERENCES

Andre, T. S., Hartson, H. R., Belz, S. M., & McCreary, F. A. (2001). The user action framework: a reliable foundation for usability engineering support tools. *International Journal of Human-Computer Studies*, *54*(1), 107–136.

Bailey, R. W., Allan, R. W., & Raiello, P. (1992). Usability testing vs. heuristic evaluation: A head-to-head comparison. *Proceedings of the Human Factors Society 36th Annual Meeting*, 409–413.

Barkhuus, L., & Rode, J. (2007, April). *From mice to men—24 years of evaluation in CHI*. Paper presented at the ACM CHI ′07—Alt.CHI, San Jose, CA. Available from http://www.viktoria.se/altchi/

Capra, M., & Smith-Jackson, T. (2005). *Developing guidelines for describing usability problems* (No. ACE/HCI-2005-002). Blacksburg: Virginia Tech, Assessment and Cognitive Ergonomics Laboratory & Human-Computer Interaction Laboratory.

Chattratichart, J., & Lindgaard, G. (2008). A comparative evaluation of heuristic-based usability inspection methods. *CHI ′08 Extended Abstracts on Human Factors in Computing Systems*, 2213–2220.

Cockton, G., & Lavery, D. (1999). A framework for usability problem extraction. *Proceedings of the IFIP TC.13 International Conference on Human-Computer Interaction 9*, 344–352.

Cockton, G., Lavery, D., & Woolrych, A. (2008). Inspection-based evaluations. In A. Sears & J. Jacko (Eds.), *The human-computer interaction handbook*, 2nd edition (pp. 1171–1190). London, UK: CRC Press.

Cockton, G., & Woolrych, A. (2001). Understanding inspection methods: Lessons from an assessment of heuristic evaluation. *People and Computers XV: Joint Proceedings of HCI 2001 and IHM 2001*, 171–192.

Cockton, G., & Woolrych, A. (2009). *Comparing UEMs: Strategies and implementation. Final report of COST-294 working group 2*. Retrieved from http://141.115.28.2/cost294/upload/533.pdf

Cockton, G., Woolrych, A., Hall, L., & Hindmarch, M. (2003). Changing analysts' tunes: The surprising impact of a new instrument for usability inspection method assessment. *People and Computers XVII: Designing for Society*, 145–162.

Cockton, G., Woolrych, A., & Hindmarch, M. (2004). Reconditioned merchandise: Extended structured report formats in usability inspection. *CHI 2004 Extended Abstracts on Human Factors in Computer Systems*, 1433–1436.

Connell, I. W., & Hammond, N. V. (1999). Comparing usability evaluation principles with heuristics: Problem instances vs. problem types. *Proceedings of the IFIP TC.13 International Conference on Human-Computer Interaction*, 621–629.

Courage, C., & Baxter, K. (2005). *Understanding your users: A practical guide to user requirements methods, tools, and techniques*. San Francisco, CA: Morgan Kaufmann.

Cuomo, D. L., & Bowen, C. D. (1992). Stages of user activity model as a basis for user-system interface evaluations. *Proceedings of the Human Factors Society 36th Annual Meeting*, 1254–1258.

Dreyfus, H., & Dreyfus, S. (1986). *Mind over machine*. New York, NY: Free Press.

Feyerabend, P. (1975). *Against method: Outline of an anarchistic theory of knowledge*. London, UK: Humanities.

Frøkjær, E., & Hornbæk, K. (2008). Metaphors of human thinking for usability inspection and design. *ACM Transactions on Computer-Human Interaction*, *14*(4), 20.

Furniss, D. (2008). *Beyond problem identification: Valuing methods in a 'system of usability practice'*. London, UK: University College London.

Furniss, D., Blandford, A., & Curzon, P. (2008). Usability work in professional website design: Insights from practitioners' perspectives. In E. Law, E. Hvannberg, & G. Cockton (Eds.), *Maturing usability: Quality in software, interaction and value* (pp. 144–167). New York, NY: Springer.

Gilbert, J. D., Williams, A., & Seals, C. D. (2007). Clustering for usability participant selection. *Journal of Usability Studies*, *3*(1), 40–52.

Gray, W. D., & Salzman, M. C. (1998). Damaged merchandise? A review of experiments that compare usability evaluation methods. *Human-Computer Interaction*, *13*, 203–261.

Gulliksen, J., Boivie, I., & Göransson, B. (2006). Usability professionals—Current practices and future development. *Interacting with Computers*, *18*, 568–600.

Hertzum, M., Hansen, K. D., & Andersen, H. H. K. (2009). Scrutinising usability evaluation: does thinking aloud affect behaviour and mental workload? *Behaviour & Information Technology*, *28*, 165–181.

Hertzum, M., & Jacobsen, N. E. (2001). The evaluator effect: A chilling fact about usability evaluation methods. *International Journal of Human-Computer Interaction*, *13*, 421–443.

Holtzblatt, K., Wendell, J. B., & Wood, S. (2005). *Rapid contextual design: A how-to guide to key techniques for user-centered design*. San Francisco, CA: Morgan Kaufmann.

Hornbæk, K. (2010). Dogmas in the assessment of usability evaluation methods. *Behaviour & Information Technology*, *29*(1), 97–111.

Hornbæk, K., & Frøkjær, E. (2004). Usability inspection by metaphors of human thinking compared to heuristic evaluation. *International Journal of Human–Computer Interaction*, *17*, 357–374.

Hornbæk, K., & Frøkjær, E. (2006). What kinds of usability-problem description are useful to developers? *Proceedings of the Human Factors and Ergonomic Society's Annual Meeting*, 2523–2527.

Hornbæk, K., & Frøkjær, E. (2008a). Comparison of techniques for matching of usability problem descriptions. *Interacting with Computers*, *20*, 505–514.

Hornbæk, K., & Frøkjær, E. (2008b). Making use of business goals in usability evaluation: an experiment with novice evaluators. *Proceedings of the ACM Conference on Human Factors in Computing Systems*, 903–912.

Howarth, J., Andre, T. S., & Hartson, R. (2007). A structured process for transforming usability data into usability information. *Journal of Usability Studies*, *3*(1), 7–23.

Hvannberg, E. T., Law, E. L.-C., & Lárusdóttir, M. K. (2007). Heuristic evaluation: Comparing ways of finding and reporting usability problems. *Interacting with Computers*, *19*, 225–240.

Jacobsen, N. E., & John, B. E. (2000). *Two case studies in using cognitive walkthroughs for interface evaluation* (Tech. Rep. No. CMU-CS-00-132). Pittsburgh, PA: Carnegie Mellon University School of Computer Science.

Jeffries, R. (1994). Usability problem reports: Helping evaluators communicate effectively with developers. In J. Nielsen & R. L. Mack (Eds.), *Usability inspection methods* (pp. 273–294). New York, NY: Wiley.

Jeffries, R., Miller, J., Wharton, C., & Uyeda, K. (1991). User interface evaluation in the real world: A comparison of four techniques. *Proceedings of ACM Conference on Human Factors in Computer Systems*, 119–124.

John, B. E., & Packer, H. (1995). Learning and using the cognitive walkthrough method: a case study approach. *Proceedings of ACM Conference on Human Factors in Computer Systems*, 429–436.

Jones, J. C. (1988). Softecnica. In J. Thackera (Ed.), *Design after modernism: Beyond the object* (pp. 216–266). London, UK: Thames & Hudson.

Jones, J. C. (1992). *Design methods: Seeds of human futures*. New York, NY: Wiley.

Kieras, D. (1988). Towards a practical GOMS model methodology for user interface design. In M. Helander (Ed.), *Handbook of human-computer interaction* (pp. 135–157). Amsterdam, the Netherlands: Elsevier Science.

Kjeldskov, J., Skov, M., & Stage, J. (2004). Instant data analysis. *Proceedings of the Third Nordic Conference on Human-Computer Interaction*, 233–240.

Lavery, D., & Cockton, G. (1997). Representing predicted and actual usability problems. *Proceedings of the International Workshop on Representations in Interactive Software Development*, 97–108.

Lavery, D., Cockton, G., & Atkinson, M. P. (1996). *Heuristic evaluation: Usability evaluation materials* (Tech. Rep. TR-1996-15). Glasgow, Scotland: University of Glasgow. Available from http://www.dcs.gla.ac.uk/asp/materials/HE_1.0/materials.pdf

Lavery, D., Cockton, G., & Atkinson, M. P. (1997). Comparison of evaluation methods using structured usability problem reports. *Behaviour and Information Technology*, *16*, 246–266.

Law, L., & Hvannberg, E. T. (2002). Complementarity and convergence of heuristic evaluation and usability test: a case study of universal brokerage platform. In *Proceedings of the Second Nordic Conference on Human-Computer interaction* (pp. 71–80). New York, NY: ACM.

Law, E., & Hvannberg, E. T. (2008). Consolidating usability problems with novice evaluators. In *Proceedings of the 5th Nordic Conference on Human-Computer interaction* (pp. 495–498). New York, NY: ACM.

Lindgaard, G., & Chattratichart, J. (2007). Usability testing: what have we overlooked? *Proceedings of the ACM Conference on Human Factors in Computing Systems*, 1415–1424.

Mankoff, J., Dey, A. K., Hsieh, G., Kientz, J., Ames, M., & Lederer, S. (2003). Heuristic evaluation of ambient displays. *Proceedings of the ACM Conference on Human Factors in Computing Systems*, 169–176.

Mathiasen, L. (1982). *Systems develpment and systems development method*. Unpublished doctoral dissertation, Aarhus University, Aarhus, Denmark.

Mayhew, D. J. (1999). *The usability engineering lifecycle*. San Francisco, CA: Morgan Kaufmann.

Molich, R., Ede, M. R., Kaasgaard, K., & Karyukin, B. (2004). Comparative usability evaluation. *Behaviour & Information Technology*, *23*(1), 65–74.

Molich, R., & Nielsen, J. (1990). Improving a human-computer dialogue. *Communications of the ACM*, *33*, 338–348.

Naur, P. (1995). *Knowing and the mystique of logic and rules*. Dordrecht, the Netherlands: Kluwer Academic.

Nielsen, J. (1994). Heuristic evaluation. In J. Nielsen & R. Mack (Eds.), *Usability inspection methods* (pp. 25–62). New York, NY: Wiley.

Nielsen, J. (n.d.). *How to conduct a heuristic evaluation*. Retrieved from http://www.useit.com/papers/heuristic/heuristic_evaluation.html

Nielsen, J., & Molich, R. (1990). Heuristic evaluation of user interfaces. *ACM Conference on Human Factors in Computing*, 249–256.

Nørgaard, M., & Hornbæk, K. (2006). What do usability evaluators do in practice? An explorative study of think-aloud testing. *ACM Conference on Designing Interactive Systems*, 209–218.

Okasha, S. (2002). *Philosophy of science: A very short introduction*. Oxford, UK: Oxford Paperbacks.

Rieman, J., Davies, S., Hair, D. C., Esemplare, M., Polson, P., & Lewis, C. (1991). An automated cognitive walkthrough. *Proceedings of the ACM Conference on Human Factors in Computing Systems*, 427–428.

Rosenbaum, S. (2008). The future of usability evaluations: Increasing impact on value. In E. L. C. Law, E. Hvannberg, & G. Cockton (Eds.), *Maturing usability: Quality in software, interaction and value* (pp. 344–378). London, UK: Springer.

Sears, A. (1997). Heuristic walkthroughs: Finding the problems without the noise. *International Journal of Human-Computer Interaction*, *9*, 213–234.

Sears, A., & Hess, D. (1999). Cognitive walkthroughs: Understanding the effect of task description detail on evaluator performance. *International Journal of Human-Computer Interaction*, *11*, 185–200.

Skov, M., & Stage, J. (2005). Supporting problem identification in usability evalautions. *Proceedings of the Australian Computer-Human Interaction Conference (OZCHI*). pp. 1–9.

Somervell, J., & McCrickard, D. S. (2005). Better discount evaluation: illustrating how critical parameters support heuristic creation. *Interacting with Computers*, *17*, 592–612.

Sorell, T. (2001). *Descartes*. Oxford, UK: Oxford University Press.

Spencer, R. (2000). The streamlined cognitive walkthrough method, working around social constraints encountered in a software development company. *Proceedings of the ACM Conference on Human Facors in Comuting Systems*, 353–359.

Spool, J., & Shroeder, W. (2001). Testing web sites: Five users is nowhere near enough. *CHI '01 Extended Abstracts on Human Factors in Computing Systems*, 285–286.

Suchman, L. (1987). *Plans and situated actions: The problem of human-machine communication*. New York, NY: Cambridge University Press.

Sy, D. (2007). Adapting usability investigations for agile user-centered design. *Journal of Usability Studies*, *2*, 112–132.

Theofanos, M., & Quesenbery, W. (2005). Towards the design of effective formative test reports. *Journal of Usability Studies*, *1*(1), 27–45.

Uldall-Espersen, T., Frøkjær, E., & Hornbæk, K. (2008). Tracing impact in a usability improvement process. *Interacting with Computers*, *20*(1), 48–63.

Vermeeren, A. P. O. S. (2009). *What's the problem? Studies on identifying usability problems in user tests*. Unpublished doctoral dissertation, Delft University of Technology, Faculty of Industrial Design Engineering, Delft, the Netherlands. Retrieved from http://homepages.ipact.nl/~vermeeren/PhDThesis%20and%20Propositions%20APOS Vermeeren-Whats%20the%20problem.pdf

Wharton, C., Rieman, J., Lewis, C., & Polson, P. (1994). The cognitive walkthrough method: A practitioner’s guide. In J. Nielsen & R. L. Mack (Eds.), *Usability inspection methods* (pp. 105–140). New York, NY: Wiley.

Wixon, D. (2003). Evaluating usability methods: Why the current literature fails the practitioner. *interactions*, *10*(4), 29–34.

Woolrych, A., & Cockton, G., (2001). Why and when five test users aren’t enough. In J. Vanderdonckt, A. Blandford, & A. Derycke (Eds.), *Proceedings of IHM-HCI 2001 Conference* (Vol. 2, pp. 105–108). Toulouse, France: Cépadèus Éditions.

Woolrych, A., Cockton, G., & Hindmarch, M. (2005). Knowledge resources in usability inspection. *British HCI Conference*, *2*, 5–20.