INTERNATIONAL JOURNAL OF HUMAN–COMPUTER INTERACTION, *17*(3), 357–374
Copyright © 2004, Lawrence Erlbaum Associates, Inc.

Usability Inspection by Metaphors of Human Thinking Compared to Heuristic Evaluation

Kasper Hornbæk
Erik Frøkjær
University of Copenhagen, Denmark

A new usability inspection technique based on metaphors of human thinking has been experimentally compared to heuristic evaluation (HE). The aim of metaphors of thinking (MOT) is to focus inspection on users' mental activity and to make inspection easily applicable to different devices and use contexts. Building on classical introspective psychology, MOT bases inspection on metaphors of habit formation, stream of thought, awareness and associations, the relation between utterances and thought, and knowing. An experiment was conducted in which 87 novices evaluated a large Web application, and its key developer assessed the problems found. Compared to HE, MOT uncovered usability problems that were assessed as more severe for users and also appeared more complex to repair. The evaluators using HE found more cosmetic problems. The time spent learning and performing an evaluation with MOT was shorter. A discussion of strengths and weaknesses of MOT and HE is provided, which shows how MOT can be an effective alternative or supplement to HE.

1. INTRODUCTION

This study entailed an experimental comparison of two usability inspection techniques. Specifically, it explored how novices identify usability problems with the aid of heuristic evaluation (HE; Molich & Nielsen, 1990; Nielsen & Mack, 1994; Nielsen & Molich, 1990), on the one hand, and with the aid of a novel technique based on metaphors of human thinking (Frøkjær & Hornbæk, 2002; Hornbæk & Frøkjær, 2002) on the other.

Inspection techniques have the aim of uncovering potential usability problems by having evaluators inspect user interfaces with a set of guidelines or questions

Both authors contributed equally to this article. We thank Peter Naur for permission to cite his writings and for invaluable support in our efforts to develop the metaphors as an inspection technique. Claus Damgaard, the key developer and manager of the Web application, did a huge effort in thoroughly assessing the usability problems, without which this study would have been impossible.

Requests for reprints should be sent to Kasper Hornbæk, University of Copenhagen, Universitetsparken 5, DK–2100 Copenhagen, Denmark. E-mail: kash@dihu.dk

(Nielsen & Mack, 1994). The most widely adopted inspection technique is HE (Rosenbaum, Rohn, & Humberg, 2000; Vredenburg, Mao, Smith, & Carey, 2002), which uses as the basis for evaluation a list of heuristics such as "be consistent" or "minimize the users' memory load" (Molich & Nielsen, 1990, p. 339). HE has been found to be a low-cost supplement to empirical techniques for identifying usability problems (Nielsen, 1994). In addition, designers may use HE and other inspection techniques throughout the design process (Shneiderman, 1998).

Despite its widespread use, HE has severe limitations. Problems identified with the aid of HE are often not found in user testing or actual use of the application—they are so-called false positives. In a recent study, more than half of the problems found were false positives (Cockton & Woolrych, 2001). HE also seems to identify many problems that pose only slight inconvenience to the user, so-called cosmetic problems (Frøkjær & Larusdóttir, 1999; Nielsen, 1992). Further, the heuristics used for evaluation are to some degree device dependent and assume a certain context of use. The heuristics used in Molich and Nielsen (1990), for example, are aimed at WIMP-style interfaces used on a desktop computer. New heuristics have had to be developed for e-commerce (Nielsen, Molich, Snyder, & Farrell, 2001) and groupware (Pinelle & Gutwin, 2002); similar efforts are found in mobile computing (Pascoe, Ryan, & Morse, 2000).

As an attempt to cope with these problems, the authors have proposed an inspection technique (Frøkjær & Hornbæk, 2002; Hornbæk & Frøkjær, 2002) based on descriptions of human thinking from classical introspective psychology (James, 1890; Naur, 1995, 2001). The technique builds on metaphoric descriptions of central aspects of human thinking—for example, habit, awareness, and associations. In addition to being independent of a certain context of use, the authors hypothesized that compared to HE this technique (a) finds more usability problems that are severe for users; (b) uncovers deeper problems, that is, problems that do not mainly concern surface-level and easily correctable problems; and (c) provides more useful input to the development process, for example, through better design ideas or through identifying problems that were previously unknown to designers.

The aim of this article is to investigate these hypotheses through an experimental comparison of the inspection technique based on metaphors of human thinking and HE. As is commonly done in experiments with usability evaluation methods, this study compares the techniques with respect to the number of problems they identify. Several studies have investigated how usability problems impact systems development (Hertzum, 1999; John & Marks, 1997; Sawyer, Flanders, & Wixon, 1996; Whiteside, Bennett, & Holtzblatt, 1988). Such analyses may lead to a more realistic and possibly different appreciation of techniques. Inspired by these studies, we examine whether the techniques differ in generating design ideas or in the severity of the problems identified as assessed by the key manager–developer of the application.

More broadly, the article aims at strengthening human–computer interaction (HCI) research and practice by making a contribution toward developing a device-independent, psychology-based evaluation method.

2. PRESENTATION OF THE INSPECTION TECHNIQUES

2.1. Heuristic Evaluation

HE, as presented by Nielsen and Molich (Molich & Nielsen, 1990; Nielsen & Molich, 1990), is based on a set of general usability principles, so-called heuristics, used to inspect user interfaces. In its most common form (Nielsen, 1993), HE consists of 10 heuristic labels: simple and natural dialogue (H1); speak the users' language (H2); minimize the users' memory load (H3); consistency (H4); feedback (H5); clearly marked exits (H6); shortcuts (H7); good error messages (H8); prevent errors (H9); help and documentation (H10).

The procedure of HE is recommended to involve a small group of evaluators. First each evaluator goes through the interface, examines all the interface elements, and judges their compliance with the 10 heuristics. For any specific interface element, the evaluator may further consider additional usability principles or circumstances that seem to be relevant. After the evaluators' individual usability inspections, the group of evaluators discusses and summarizes their results to reach a more comprehensive report of the usability problems of the interface.

2.2. Evaluation by Metaphors of Human Thinking

Compared to HE, the aim of metaphors of thinking (MOT) is to focus inspection on users' mental activity through metaphors inspired by classical introspective psychology. This section summarizes MOT by describing the five supporting metaphors and the underlying understanding of human thinking. For each metaphor, a few key questions to consider in a usability inspection and an example of its use in design are given. Metaphors in the HCI literature have been used in describing certain styles of interfaces, for example, the desktop metaphor (Johnson et al., 1989), and as a vehicle for representing and developing designs of interfaces (Erickson, 1990; Madsen, 1994; Neale & Carroll, 1997). This article uses the term *metaphors* differently, in that the metaphors are not in any way intended as interface metaphors, nor are the metaphors imagined to form part of designs. Rather, the purpose of the metaphors is to support the evaluator–systems designer in a focused study of how well certain important aspects of human thinking are taken into account in the user interface under inspection. The metaphors are intended to stimulate thinking, generate insight, and break fixed conceptions. These uses of metaphors have been thoroughly studied in the literature on creative thinking (Gardner, 1982; Kogan, 1983) and illustratively applied, for example, by Sfard (1998) in the educational domain.

Metaphor M1: Habit formation is like a landscape eroded by water. Habits shape most of human thought activity and behavior (e.g., as physical habits, automaticity, all linguistic activity, and habits of reasoning). This metaphor should indicate how a person's formation of habits leads to more efficient actions and less conscious effort, just as a landscape, through erosion, adapts for a more efficient and smooth flow of water. Creeks and rivers will, depending on changes in water flow, find new ways or become arid and sand up, in the same way as a person's habits will adjust to new circumstances and, if unpracticed, vanish. Usability inspection with M1 calls for these considerations: Are existing habits supported? Can effective new habits, when necessary or appropriate, be developed? Can the user use common key combinations? Is it possible for the user to predict—a requisite for forming habits—the layout and functioning of the interface?

In design, there is an abundance of examples of user interfaces that prevent habit formation. One example is adaptive menus, used, for example, in Microsoft Office 2000. Adaptive menus change the layout of the menu according to how often menu items are used—for example, by removing or changing the position of items seldom used. However, adaptive menus make it impossible to form habits in the selection of menu items, because their position may be different from when they were previously selected. A study by Somberg (1987) showed the efficiency of constant position placement of menu items compared to menus that change based on use frequency. Somberg, however, did not explicitly link habit formation to the usefulness of constant placement of menu items.

Metaphor M2: Thinking as a stream of thought. Human thinking is experienced as a stream of thought—for example, in the continuity of people's thinking, and in the richness and wholeness of a person's mental objects, of consciousness, of emotions and subjective life. This metaphor was proposed by James (1890, Vol. 1) to emphasize how consciousness does not appear to itself be chopped up in bits: "Such words as 'chain' or 'train' do not describe it fitly. It is nothing jointed; it flows" (p. 239). Particular issues can be distinguished and retained in a person's stream of thought with a sense of sameness, as anchor points, which function as "the keel and backbone of human thinking" (James, 1890, Vol. 1, p. 459). Usability inspection with M2 calls for these considerations: Is the flow of users' thoughts supported in the interface by recognizability, stability, and continuity? Does the application make visible and easily accessible interface elements that relate to the anchor points of users' thinking about their tasks? Does the application help users to resume interrupted tasks?

In design, a simple, yet effective, attempt to recreate part of the richness of the stream of thought when users return to resume interrupted work is Raskin's (2000) design of the Canon Cat. When the Canon Cat is started, the display immediately shows up as it was before work was suspended. Not only does this allow the user to start thinking about the task at hand while the system is booting but it also provides help in remembering and recreating the stream of thought as it was when work was interrupted.

Metaphor M3: Awareness as a jumping octopus in a pile of rags. Here the dynamics of human thinking are considered, that is, the awareness shaped through a focus of attention, the fringes of mental objects, association, and reasoning. This metaphor was proposed by Peter Naur (1995, pp. 214–215) to indicate how the state of thought at any moment has a field of central awareness, that part of the rag pile in which the body of the octopus is located; but at the same time it has a fringe of vague and shifting connections and feelings, illustrated by the arms of the octopus stretching out into other parts of the rag pile. The jumping about of the octopus indicates how the state of human thinking changes from one moment to the next. Usability inspection with M3 calls for these considerations: Are users' associations supported through flexible means of focusing within a stable context? Do users associate interface elements with the actions and objects they represent? Can words in the interface be expected to create useful associations for the user? Can the user switch flexibly between different parts of the interface?

In design, an example of a problematic solution in light of this metaphor is a use of modal dialog boxes that prevents the user from switching to potentially relevant information—in Microsoft Word, for example, it is not possible to switch back to the document to look for a good file name once the "save as" dialog has begun.

Metaphor M4: Utterances as splashes over water. Here the focus is on the incompleteness of utterances in relation to the thinking underlying them and the ephemeral character of those utterances. This metaphor was proposed by Naur (1995, pp. 214–215) to emphasize how utterances are incomplete expressions of the complexity of a person's current mental object, in the same way as the splashes over the waves tell little about the rolling sea below. Usability inspection with M4 calls for these considerations: Are changing and incomplete utterances supported by the interface? Are alternative ways of expressing the same information available? Are the interpretations of users' input in the application made clear? Does the application make a wider interpretation of users' input than users intend or are aware of?

For design, one implication of the metaphor of utterances as splashes over the waves is that users must be expected to describe the same objects and functions incompletely and in a variety of ways. Furnas, Landauer, Gomez, and Dumais (1987) investigated the diversity in words used for describing commands and everyday objects. On the average, 2 participants described the same command or object by the same term with less than 20% probability. The most popular name was chosen only in 15% to 35% of the cases. Furnas et al.'s suggestion for relieving this problem is called the *unlimited alias approach*, where terms unknown to the system may be interactively related to existing commands or object names. This solution is coherent with the metaphor and uses interactivity to clarify the intentions of the user. However, it would partly go against the metaphor of habit formation.

Metaphor M5: Knowing as a building site in progress. Human knowing is always under construction and incomplete. This metaphor was also proposed by Naur (1995, pp. 214–215) and was meant to indicate the mixture of order and inconsistency characterizing any person's insight. These insights group themselves in many ways, the groups being mutually dependent by many degrees, some closely, some slightly. As an incomplete building may be used as shelter, so the insights had by a person in any particular field may be useful even if restricted in scope. Usability inspection with M5 calls for these considerations: Are users forced by the application to depend on complete or accurate knowledge? Is it required that users pay special attention to technical or configuration details before beginning to work? Do more complex tasks build on the knowledge users have acquired from simpler tasks? Are users supported in remembering and understanding information in the application?

In design, mental models have been extensively discussed. Consider as an example Norman's (1983) description of the use of calculators. He argued that the use of calculators is characterized by users' incomplete understanding of calculators, by the instability of the understanding, by superstitions about how calculators work, and by the lack of boundaries in the users' understanding of one calculator and another. These observations by Norman are perfectly coherent with the ideas expressed by the metaphor of knowing.

Procedure for performing a MOT evaluation. The basic procedure when using the metaphors for evaluating user interfaces is to inspect the interface, noting when it supports or violates the aspects of human thinking that the metaphors and key questions aim to capture. This enables the evaluators to identify potential usability problems. The evaluation will result in a list of usability problems, each described with reference to the application and to metaphors that were used to uncover the problem. The usability problems may then be given a severity rating, and suggestions may be made as to how to correct the problem. In Hornbæk and Frøkjær (2002), each of the metaphors and their implications for user interfaces are described in more detail, and a procedure of how to do a usability inspection based on these metaphors is proposed. This procedure is quite similar to the one described earlier for HE.

The steps in the procedure are as follows:

1. Familiarize yourself with the application.
2. Find three tasks that users typically would do with the application. These tasks may be thought up, may be based on observations of users, or may be based on scenarios used in systems development.
3. Try to do the tasks with the application. Identify major problems found in this way. Use the key questions and the metaphors to find usability problems.
4. Do the tasks again. This time, take the perspective of each of the metaphors, one at a time, and work through the tasks. Use the key questions and the metaphors to find usability problems.
5. If more time is left, find some more tasks. See if new problems arise in Steps 3 and 4 for those tasks. Iterate Steps 3 and 4 until each new task reveals few new problems or until no time is left.

MOT differs, however, from HE in one crucial aspect: HE is aimed toward providing simple guidelines with straightforward interpretations, but MOT provides guidelines that are complex and require evaluators' active interpretation. Although the techniques thus appear very different, no studies have empirically evaluated their differences, nor do data exist as to whether MOT can be easily understood and applied. These observations provide the rationale for the experiment described next.

3. METHOD

In the experiment, participants using either MOT or HE inspected a Web application. The problems found were consolidated to a common list, and the key manager–developer of the application assessed the problems.

3.1. Objectives

The experiment had as its objectives to compare MOT and HE by (a) the number of problems they identified, (b) how evaluators assessed the techniques, and (c) how the key developer of the application assessed the usability problems found.

3.2. Application

The Web application inspected is a portal for students at the University of Copenhagen to course administration, e-mail, information on grades, university news, and so forth (see http://punkt.ku.dk). The application builds on and exchanges data with five existing administrative systems. The version of the application inspected took approximately five person-months to develop.

3.3. Participants

As a compulsory part of a 1st-year university course in multimedia technology, 87 computer science students used either HE or MOT to evaluate the Web application. Participation was anonymous, and the students were free to choose whether their data could be included in the analysis.

3.4. Procedure for Participants' Inspection

Forty-four participants received as a description of MOT a pseudonymized version of Hornbæk and Frøkjær (2002); 43 participants received as a description of HE pp. 19 to 20 and 115 to 163 from Nielsen (1993). Each participant individually performed the evaluation supported by scenarios made available by the developers of the Web application. The participants were instructed to write for each usability problem identified (a) a brief title, (b) a detailed description, (c) an identification of the metaphors or heuristics that helped uncover the problem, and (d) a seriousness rating.

Participants chose seriousness ratings from a commonly used scale (Molich, 1994, p. 111): *Rate 1* is given to a critical problem that gives rise to frequent catastrophes that should be corrected before the system is put into use. This grade is for those few problems that are so serious that the user is better served by a delay in the delivery of the system. *Rate 2* is given to a serious problem that occasionally gives rise to catastrophes that should be corrected in the next version. *Rate 3* is given to a cosmetic problem that should be corrected sometime when an opportunity arises.

3.5. Consolidation of Problems

To find problems that were similar to each other, a consolidation of the problems was undertaken. In this consolidation, we grouped together problems perceived as alike. The consolidation was done over a 5-day period, with at least two passes over each problem. The consolidation was done blind to what technique had produced the problems.

This consolidation resulted in a list of 341 consolidated problems. Each consolidated problem consisted of one or more problems. Figure 1 gives a preview of the relation between problems and consolidated problems; the Results section treats this relation in detail.

To test the reliability of the consolidation, an independent rater tried consolidating a random subset of the problems. The rater received 53 problems together with the list of the consolidated problems from which these 53 problems had been deleted. For each problem, the rater either grouped together that problem with a con-

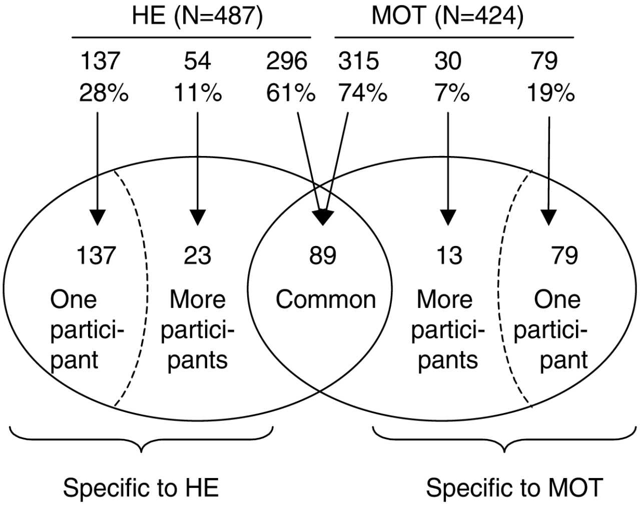

FIGURE 1 The relation between problems and consolidated problems. The top part of the figure shows the problems identified with the two techniques. The circles at the bottom part of the figure show the consolidated problems. The arrows indicate the number and percentage of problems in each group of consolidated problems. HE = heuristic evaluation; MOT = metaphors of thinking.

solidated problem or noted that the problem was not similar to any of the consolidated problems. Using Cohen's kappa, the interrater reliability between ratings was .77, suggesting an excellent agreement beyond chance (Fleiss, 1981).

3.6. The Client's Assessment of Problems

This study entailed trying to make a realistic assessment of usability problems—similar to what goes on when managers and developers read and prioritize problems in usability reports. The person representing the client in this study is the person who decides what to do about the problems. This approach to assessment is close to what has been called the *impact ratio of inspections* (Sawyer et al., 1996), which entails a similar, pragmatic view of how to assess usability problems. A related approach was suggested by Hartson, Andre, and Williges (2001):

> Perhaps an alternative way to evaluate post-usability testing utility of UEM [usability evaluation methods] outputs is by asking real-world usability practitioners to rate their perceptions of usefulness of problem reports in meeting their analysis and redesign needs within the development cycle for interaction design in their own real development environment.

Furthermore, the realistic assessment of usability problems comes closer to meeting the challenge put forward by Wixon (2003) of going beyond using counts of problems to compare techniques.

Developers have a vested interest in minimizing redesign to meet time and cost constraints and thus may be inherently biased in their assessment of usability problems. However, in practice these are the circumstances that determine which problems are addressed and how they are addressed. Thus, a crucial part of the experiment was having the consolidated problems assessed by persons who were developing the Web application, here called the *client*. In this experiment, the person who managed the development of the Web application and was responsible for developing the design represented the client. The usability problems found have been used in the client's development of a new version of the application.

For each consolidated problem, the client was asked to assess aspects considered likely to influence how to prioritize and revise the design. The aspects considered were as follows:

- Severity of the problem. The severity of the problem, related to users' ability to do their tasks, was judged as 1 (*very critical problem*), 2 (*serious problem*), 3 (*cosmetic problem*), or % (*not a problem*). Note that this grading is different from the students' seriousness ratings in that only the nature of the problem is being assessed, not when the problem should be corrected, which is contingent on resources within the development organization.
- Design ideas generated by the problem. The client stated (yes or no) whether the development team, from the problem descriptions, got any good ideas for how to resolve the problem.

- The novelty of the problem. The client stated (yes or no) whether the problem brought up a new or surprising difficulty with the Web application.
- The perceived complexity of solving the problem. The client also assessed how complex it would be to reach a clear and coherent proposal for how to change the Web application so as to remove the problem. The client used a four-rating scale to judge complexity: 1 = *very complex problem*: takes several weeks to make a new design, possibly involving outside expert assistance; 2 = *complex problem*: a suggestion may be arrived at by experts in the development group in a few weeks; 3 = *middle complexity*: new design can be devised in a few days; 4 = *simple*: while the actual implementation may take longer, a solution to the problem can be found in a few hours.

The assessment was done from a list, which for each consolidated problem showed all the problems that it was consolidated from. The client performed the rating blind to what technique had produced the problems, and he was not familiar with what techniques were studied.

4. RESULTS

Table 1 summarizes the differences in problems between techniques; Table 2 shows the overlap between problems as determined by the consolidation of problems. An overall multivariate analysis of variance on these data showed a significant difference between techniques, Wilks's $\Lambda = .715$, $F(8, 78) = 3.88$, $p < .001$. With this test protecting the experiment-wise error, the data were analyzed from the two tables with individual analyses of variance. Note that ratings and other ordinal data were

Table 1: The Client's Assessment of Usability Problems Found by Participants Using Either Heuristic Evaluation (HE) or Evaluation by Metaphors of Human Thinking (MOT)

	HE (n = 43)			*MOT (n = 44)*		
	Mean	*SD*	*%*	*Mean*	*SD*	*%*
No. of problems	11.3	6.2	—	9.6	5.7	—
Severity (average)***	2.4	0.9	—	2.2	0.7	—
Very critical (1)	0.8	1.1	7	1.2	1.1	12
Serious (2)	4.8	3.0	42	5.0	3.6	52
Cosmetic (3)	5.6	4.2	49	3.2	2.8	33
Not a problem (%)	0.1	0.4	1	0.3	0.5	3
Complexity (average)***	3.2	1.0	—	3.00	0.8	—
Very complex (1)	0.1	0.3	1	0.02	0.2	0
Complex (2)	2.7	1.9	24	3.3	2.5	34
Middle complexity (3)	2.8	2.0	24	2.3	1.9	23
Simple (4)	5.2	3.7	46	3.3	2.6	35
Not graded (%)	0.5	0.8	5	0.7	0.9	7
Novel problems***	3.8	2.8	34	2.0	1.5	28
Design ideas	2.5	1.9	22	2.2	2.2	23

Note. Averages are weighted by the number of problems; HE = heuristic evaluation; MOT = evaluation by metaphors of thinking. Due to rounding errors percentages may not add up.

***$p < .001$.

Table 2: Overlap Between Techniques and Participants in Problems Found

	HE (n = 43)			*MOT (n = 44)*		
	Mean	*SD*	*%*	*Mean*	*SD*	*%*
No. of problems	11.3	6.2	—	9.6	5.7	—
Found by both techniques	6.9	3.6	61	7.2	4.3	74
Found with one technique						
Many participants	1.3	1.4	11	0.7	1.2	7
One participant*	3.2	3.0	28	1.8	1.8	19

Note. HE = heuristic evaluation; MOT = evaluation by metaphors of thinking.
$*p < .05$.

analyzed with parametric tests, justified by the large number of observations; however, all results have been corroborated with nonparametric tests.

4.1. Number of Problems and Participants' Seriousness Rating

There was no significant difference between the number of problems participants identified with the two techniques, $F(1, 85) = 1.76$, $p > .1$. Between participants, large differences exist in the number of problems uncovered. For example, 1 participant found only 2 problems; another found 28.

Participants' ratings of the seriousness of the problems found differed only marginally between techniques, $F(1, 85) = 2.98$, $p = .09$. Problems found by participants using MOT ($M = 2.14$, $SD = 1.31$) were reported as marginally more serious than were problems found by HE ($M = 2.28$, $SD = 1.05$).

4.2. Client's Assessment

When we analyzed the client's assessment of the severity of problems, a significant difference between techniques was found, $F(1, 85) = 15.51$, $p < .001$. The client assessed problems identified with MOT as more severe ($M = 2.21$, $SD = 0.73$) than problems identified by HE ($M = 2.42$, $SD = 0.87$). As can be seen from Table 1, 49% of the problems identified with HE were assessed as being cosmetic problems by the client; only 33% of the problems found with MOT were assessed as being cosmetic. The number of problems that the client did not perceive as usability problems was surprisingly small, between 1% and 3%, compared to what other studies have reported. As mentioned earlier, Cockton and Woolrych (2001) found more than half of the problems identified with HE to be false positives. In Frøkjær and Larusdóttir (1999), HE stood out by identifying 23% false problems compared to cognitive walk-through (0%) and thinking aloud (1%).

The client's assessment of the number of design ideas to be gained from the problem descriptions was not significantly different between techniques, $F(1 ,85) = 0.35$, $p > .5$.

Concerning the number of novel problems, HE identified significantly more than MOT, $F(1, 85) = 14.59$, $p < .001$. On average, participants using HE found 3.8

(*SD* = 2.8) problems that the client considered novel; participants using MOT found 2.0 (*SD* = 1.5) such problems on average. In interpreting this difference, it seems that novel problems may be of two kinds, either severe problems that the client had previously overlooked or cosmetic and somewhat esoteric problems found by only 1 participant. For both techniques, novel problems were mostly of the second kind, as novel problems, on average, were less severe (*M* = 2.31, *SD* = 0.75) and less complex (*M* = 3.48, *SD* = 0.71), and 41% were only found by 1 participant.

The complexity of the problems identified was significantly different between techniques, $F(1, 85) = 12.94$, $p < .001$. The client assessed problems found with MOT as more complex to solve (*M* = 3.00, *SD* = 0.80) compared to those found by HE (*M* = 3.21, *SD* = 0.96). As shown in Table 1, approximately 20% more problems considered "complex" were found with MOT compared to HE; around 60% more problems considered "simple" were found with HE compared to MOT.

Note that severity and complexity are correlated ($r_s = .40$, $p < .001$). This suggests that in this study severity and complexity cannot be regarded as being independent.

The data from the client's assessment are supported by the marginally significant difference in participants' ratings of problem seriousness, where problems found by MOT were rated as more serious.

4.3. Overlap Between Evaluators and Techniques

One use of the consolidation of problems is to describe the overlap between participants using the same technique and the overlap between techniques (see Table 2 and Figure 1).

Between techniques, a significant difference was found in the number of problems identified by only 1 participant, $F(1, 85) = 6.58$, $p < .05$. On average, participants using HE found 78% more 1-participant problems compared to participants using MOT. Incidentally, the 1-participant problems found by MOT and those found by HE have comparable low (2.72) average severity (MOT: *SD* = 0.80; HE: *SD* = 0.62). As further related to the number of 1-participant problems, participants using MOT found problems that were more generally agreed upon among the participants as usability problems. When a measure from research looking at the evaluator effect (Hertzum & Jacobsen, 2001) was used, the average overlap in problems found by two evaluators using MOT—the so-called any-two agreement measure—was 9.2%, whereas the any-two agreement for HE was 7.2%.

HE identified 74% of the problems identified by MOT; MOT identified 61% of the problems identified by HE. The large number of 1-participant problems identified by HE resulted in the total number of different problems identified being larger for HE (249), compared to MOT (181). Thus, in some sense, HE resulted in a broader class of problems.

4.4. Use of Individual Metaphors and Heuristics

A central question with both MOT and HE is the utility of individual metaphors and heuristics in helping evaluators predict usability problems. Table 3 shows

Table 3: Relation Between and Use of Individual Heuristics and Metaphors

	No. of Problems	*Percentage Found By Only 1 Participant*	*Average Severity*	*Heuristic–Metaphor Also Identifying Problems*
Heuristic	800	30	2.4	
H1	205	30	2.5	H2 (52%); M3 (49%)
H2	117	25	2.5	H1 (74%); H4 (56%)
H3	63	29	2.4	H1 (70%); M2 (58%)
H4	134	24	2.3	H1 (67%); M3 (54%)
H5	43	35	2.3	H1 (42%); M2 (28%)
H6	41	34	2.5	H4 (55%); M2 (41%)
H7	28	36	2.2	H1, H3, M3 (57%)
H8	41	37	2.0	H9 (63%); H2 (49%)
H9	64	28	2.2	H1 (56%); H2, M1 (41%)
H10	64	36	2.6	H1 (73%); H2 (55%)
Metaphor	573	18	2.2	
M1	124	16	2.2	M2 (69%); M3 (67%)
M2	169	21	2.2	M3 (74%); M1 (66%)
M3	141	16	2.3	M2 (75%); M1 (63%)
M4	66	21	2.4	M3 (59%); M2 (52%)
M5	73	19	2.0	M2 (66%); M1 (64%)

Note. H1 to H10 refers to the heuristics mentioned in Nielsen (1993); M1 to M5 refer to the metaphors described in Hornbæk and Frøkjær (2002). (Section 2 describes these heuristics and metaphors.) Severity was measured on a 1-to-3 scale, with 1 being the most severe. Participants could indicate more than one metaphor per problem, which is why the sum of column 2 is higher than the number of problems. The rightmost column shows the heuristics–metaphors with the two largest overlaps with the heuristic–metaphor in the row.

some initial data that might be helpful in exploring this question in the context of novice evaluators and the particular Web application evaluated.

For the heuristics, large differences exist in the number of problems identified. Heuristic H1 (simple and natural dialogue) helped participants find 205 problems, 62 of which were single-participant problems; H7 (shortcuts) identified only 28 problems. In this study, H8 (good error messages) stands out as particularly good at identifying severe problems. Previously, Nielsen (1992) showed that evaluators have difficulty in applying slightly different versions of heuristics H6, H8, and H9. Also in this study, H6 (clearly marked exits) seems difficult to use, in that it identified problems with a low severity rating. H1 contributed the highest absolute number of 1-participant problems among any heuristic–metaphor. This high contribution perhaps suggests that H1 is understood and applied very differently among participants, though in part this result may have been produced artificially by the ordering of heuristics.

Each metaphor has less variation in the number of problems found compared to the heuristics. M5 (knowing as a building site in progress) seems to help participants the most in finding severe usability problems. M4 (utterances as splashes over water) identified fewer problems than the other metaphors, and the problems found were also slightly less severe.

Comparing heuristics and metaphors, it is noteworthy that problems found with metaphors were most frequently found with more than one metaphor (see Table 3). Problems found with HE frequently overlapped with problems found with the metaphors.

4.5. Learning and Usage Experiences

For reading and performing the inspections, the participants reported spending, on average, 4.0 hr for MOT (*SD* = 2.3) and 5.8 hr for HE (*SD* = 3.8). This difference was significant (Mann–Whitney $U = 546.5$, $z = -2.88$, $p < .01$). However, the descriptions of the two inspection techniques were of quite different lengths. The MOT description contained 10,500 words, and the HE description approximately 17,000 words. As participants were not asked to distinguish between time spent reading and time spent performing the evaluation, it can just be noted that participants reported spending less time using MOT; this result does not point to why.

The participants wrote in groups of 3 to 4 persons about their experiences with learning and using the inspection technique that they had tried. These reports provide a main impression of MOT as being more difficult to learn. The task of understanding the ideas about human thinking and the metaphors seems to have been quite demanding for participants, but participants reported that the examples and key questions to consider during an inspection (see Hornbæk & Frøkjær, 2002) were helpful. The description of HE was generally acknowledged as being well presented. Note that none of the techniques were introduced orally so as to ensure that the participants were uninformed about the project's specific interest in MOT. The participants were familiar with Shneiderman's (1998) eight golden rules of interface design (pp. 74–75) from lectures and the course textbook. This familiarity may have served partly as an introduction to HE.

Both for HE and for MOT, many groups told of problems with choosing and referring heuristics or metaphors to a specific usability problem. For some of the users of MOT, it seemed to have caused difficulty that some of the metaphors were tightly related. This difficulty was mentioned less often for the heuristics. For HE it was not always easy to find a relevant heuristic, for instance, if functionality was missing. HE participants often mentioned that they felt that some of the problems identified were mainly a matter of taste. Some groups acknowledged that the metaphors supported an understanding not only of usability problems in an interface but also of possible elegant design solutions.

5. DISCUSSION

Concerning the first hypothesis, the experiment showed that MOT, compared to HE, identified problems that were assessed by the client as more severe for users. In addition, participants using MOT found fewer 1-participant problems, which across techniques were assessed by the client as less severe. Concerning the second hypothesis, problems found with MOT were assessed, on average, as more complex to repair, suggesting that they go deeper than problems found with HE. Concerning the third hypothesis, the research did not indicate any difference between techniques in the number of design ideas the client got from the descriptions of problems. However, HE resulted in more problems that were assessed as novel and surprising. In this way, HE identified a broader class of problems, although these problems were mostly 1-participant problems assessed by the client to be cosmetic. Concerning learning of the techniques, participants seemed equally able to pick up the techniques, and they experienced the time needed to learn and perform an evaluation with MOT as shorter but more difficult.

Overall, the experiment shows inspection by metaphors of human thinking as a promising alternative and supplement to HE. This initial result is encouraging, because HE has been refined for many years and consistently has performed well when compared to other inspection techniques. Validation of the results in further experiments is needed to address some of the limitations of and questions raised by the experiment, two of which we now discuss.

First, in this experiment only novices' use of MOT was studied. Previous inspection-technique studies, for example, Nielsen (1992), have shown that more experienced evaluators find more problems and that the kinds of problems found are also likely to be different. It is not known if HCI persons in industry, for instance, will react to and use MOT differently than did our participants. However, novices are an important audience for effective inspection techniques. Each year, thousands of computer science students need to receive a first, and perhaps their only, introduction to evaluation of user interfaces.

Second, in this experiment only one application was evaluated. Although this application shares some characteristics with many other Web applications, one cannot conclude from this study that MOT leads to better inspections for all applications. The experiment does not shed light on whether MOT is more applicable to new devices or use contexts. An experiment addressing this question would necessarily place HE at a disadvantage, because at least some of the heuristics are closely associated with particular use contexts and interaction styles.

An important issue is how to improve MOT and HE based on the results from the experiment. For the metaphors, more examples are needed of how non-WIMP interfaces violate or respect aspects of thinking as captured by the metaphors. For the heuristics, it seems (based on the usage of individual heuristics) that perhaps a broader formulation of heuristic H7 (shortcuts), for example, is needed when Web applications are inspected.

The techniques could be combined so as to uncover a broader range of problems, because there is limited overlap between the problems found with MOT and HE (see Figure 1). Perhaps other usability evaluation methods might be more complementary in finding different problems, for example, HE and think aloud, which Frøkjær and Larusdóttir (1999) found to be a very useful combination. In future work, the qualitative differences in the problems found have to be explored in more detail. Do they concern different kinds of usability problems? And if so, is there utility in combining the methods?

Alternatives to the way the severity of problems was evaluated—for example, using think-aloud tests as a gold standard—have their own limitations. For think-aloud tests, it is difficult to extract problems from test data in a structured way (Cockton & Lavery, 1999). However, the pragmatic assessment of usability problems herein may be extended still further. For example, during a development project, one could study for all or most of the project how usability problems identified by different techniques are treated—or put aside. A possible unexplored strength of MOT is the utility of the technique in design—for example, with experienced designers as the evaluators participating in real design and development processes, essentially exploring more fully the third hypothesis mentioned in the Introduction. This experiment was not designed to make an extensive assessment of this hypothesis.

Finally, it would be interesting to investigate in more detail how the metaphors are applied during evaluation. For example, a study of evaluators thinking aloud could be conducted.

6. CONCLUSION

A new usability inspection technique based on metaphors of human thinking has been experimentally compared to HE. MOT focuses inspection on users' mental activity through five metaphors of essential aspects of human thinking. The experiment showed that MOT, compared to HE, uncovered more of the usability problems that were assessed by the key developer and manager to be severe for users and complex to repair. In addition, the evaluators using MOT showed stronger agreement, finding the same problems more often, and used less time in performing their evaluation but found MOT more difficult to learn.

It is remarkable how MOT in this first experimental study has given good results compared to HE, the usability inspection technique most widely used in industry. HE usually performs very well in comparison with other inspection techniques, for example, cognitive walk-through and GOMS-based techniques. It must be emphasized that these results have to stand up to the challenges of further study. What happens when MOT is used for evaluating interfaces in nontraditional use contexts, when the evaluators are more proficient, or when MOT is used in design work?

REFERENCES

Cockton, G., & Lavery, D. (1999). A framework for usability problem extraction. In M. A. Sasse & C. Johnson (Eds.), *Seventh IFIP Conference on Human–Computer Interaction Incorporating HCI '99* (pp. 344–352). Amsterdam: IOS Press.

Cockton, G., & Woolrych, A. (2001). Understanding inspection methods: Lessons from an assessment of heuristic evaluation. In A. Blandford, J. Vanderdonckt, & P. D. Gray (Eds.), *People and computers: Vol. 15. Joint proceedings of HCI 2001 and IHM 2001* (pp. 171–192). Berlin: Springer-Verlag.

Erickson, T. D. (1990). Working with interface metaphors. In B. Laurel (Ed.), *The art of human computer interface design* (pp. 65–73). Reading, MA: Addison-Wesley.

Fleiss, J. L. (1981). *Statistical methods for rates and proportions*. New York: Wiley.

Frøkjær, E., & Hornbæk, K. (2002). Metaphors of human thinking in HCI: Habit, stream of thought, awareness, utterance, and knowing. In R. Kuchinsky, L. Johnson, & F. Vetere (Eds.), *Proceedings of HF2002/OzCHI 2002* (CD-Rom). Australia: CHISIG.

Frøkjær, E., & Larusdóttir, M. (1999). Predicting of usability: Comparing method combinations. In M. Khosrowpour (Ed.), *Managing Information Technology Resources in Organizations in the New Millenium* (pp. 248–257) Hershey, PA: Idea Group.

Furnas, G. W., Landauer, T. K., Gomez, L. M., & Dumais, S. T. (1987). The vocabulary problem in human–system communication. *Communications of the ACM, 30*, 964–971.

Gardner, H. (1982). *Art, mind and brain: A cognitive approach to creativity*. New York: Basic Books.

Hartson, H. R., Andre, T. S., & Williges, R. C. (2001). Criteria for evaluating usability evaluation methods. *International Journal of Human–Computer Interaction, 13,* 373–410.

Hertzum, M. (1999). User testing in industry: A case study of laboratory, workshop, and field tests. In A. Kobsa & C. Stephanidis (Eds.), *User interfaces for all: Proceedings of the 5th ERCIM workshop* (pp. 59–72). Sankt Augustin, Germany: GMD.

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.

Hornbæk, K., & Frøkjær, E. (2002). Evaluating user interfaces with metaphors of human thinking. In N. Carbonell & C. Stephanidis (Eds.), *Universal access: Theoretical perspectives, practice, and experience. Lecture note in computer science 2615* (pp. 486–507). Berlin: Springer Verlag.

James, W. (1890). *Principles of psychology* (Vol. 1). New York: Holt.

John, B. E., & Marks, S. J. (1997). Tracking the effectiveness of usability evaluation methods. *Behaviour and Information Technology, 16*(4/5), 188–202.

Johnson, J., Roberts, T., Verplank, W., Smith, D., Irby, C., Bear, M., et al. (1989). The Xerox star: A retrospective. *IEEE Computer, 22*(9), 11–29.

Kogan, N. (1983). Stylistic variation in childhood and adolescence: Creativity, metaphor, and cognitive styles. In J. H. Flavell & E. M. Markman (Eds.), *Handbook of child psychology: Vol. 3. Cognitive development* (pp. 630–705). New York: Wiley.

Madsen, K. H. (1994). A guide to metaphorical design. *Communications of the ACM*, *37*(12), 57–62.

Molich, R. (1994). *Brugervenlige edb-systemer* [User friendly computer systems]. Copenhagen, Denmark: Teknisk Forlag.

Molich, R., & Nielsen, J. (1990). Improving a human–computer dialogue. *Communications of the ACM, 33*(3), 338–348.

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

Naur, P. (2001). *Anti-philosophical dictionary*. Gentafe, Denmark: Naur.com Publishing.

Neale, D. C., & Carroll, J. M. (1997). The role of metaphors in user interface design. In M. G. Helander, T. K. Landauer, & P. V. Prabhu (Eds.), *Handbook of human–computer interaction* (2nd ed., pp. 441–462). Amsterdam: Elsevier.

Nielsen, J. (1992). Finding usability problems through heuristic evaluation. In P. Baursfield, J. Bennet, & G. Lynch (Eds.), *ACM CHI'92 Conference on Human Factors in Computing Systems* (pp. 373–380). New York: ACM Press.

Nielsen, J. (1993). *Usability engineering*. San Diego, CA: Academic.

Nielsen, J. (1994). Guerrilla HCI: Using discount usability engineering to penetrate the intimidation barrier. In R. G. Bias & D. J. Mayhew (Eds.), *Cost-justifying usability* (pp. 245–272). Boston: Academic.

Nielsen, J., & Mack, R. L. (1994). *Usability inspection methods*. New York: Wiley.

Nielsen, J., & Molich, R. (1990). Heuristic evaluation of user interfaces. In J. C. Chew & J. Whiteside (Eds.), *Proceedings of the ACM Conference on Human Factors in Computing* (pp. 249–256). New York: ACM Press.

Nielsen, J., Molich, R., Snyder, C., & Farrell, S. (2001). *E-commerce user experience*. Fremont, CA: Nielsen Norman Group.

Norman, D. (1983). Observations on mental models. In D. Gentner & A. L. Stevens (Eds.), *Mental models* (pp. 7–14). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Pascoe, J., Ryan, N., & Morse, D. (2000). Using while moving: HCI issues in fieldwork environments. *ACM Transactions on Computer–Human Interaction, 7*, 417–437.

Pinelle, D., & Gutwin, C. (2002). Groupware walkthrough: Adding context to groupware usability evaluation. In D. Wixon & L. Terveen (Eds.), *ACM Conference on Human Factors in Computing* (pp. 455–462). New York: ACM Press.

Raskin, J. (2000). *The humane interface: New directions for designing interactive systems*. Reading, MA: Addison-Wesley.

Rosenbaum, S., Rohn, J., & Humberg, J. (2000). A toolkit for strategic usability: Results from workshops, panels, and surveys. In T. Turner & G. Szwillus (Eds.), *Proceedings of the CHI 2000 Conference on Human Factors in Computing Systems* (pp. 337–344). New York: ACM Press.

Sawyer, P., Flanders, A., & Wixon, D. (1996). Making a difference—The impact of inspections. In M. J. Tauber (Ed.), *ACM Conference on Human Factors in Computing* (pp. 376–382). New York: ACM Press.

Sfard, A. (1998). On two metaphors for learning and on the dangers of choosing just one. *Educational Researcher, 27*(2), 4–13.

Shneiderman, B. (1998). *Designing the user interface* (3rd ed.) Reading, MA: Addison-Wesley.

Somberg, B. L. (1987). A comparison of rule-based and positionally constant arrangements of computer menu items. In J. Carroll & P. T. Tanner (Eds.), *Proceedings of CHI+GI '87* (pp. 255–260). New York: ACM Press.

Vredenburg, K., Mao, J.-Y., Smith, P. W., & Carey, T. (2002). A survey of user-centered design practice. In L. Terveen, D. Wixon, E. Comstock, & A. Sasse (Eds.), *Proceedings of ACM Conference on Human Factors in Computing Systems* (pp. 472–478). New York: ACM Press.

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

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