7th ERCIM Workshop on User Interfaces for All, Paris 24th-25th October 2002, also in Lecture Notes in Computer Science, Vol. 2615, pp. 486-507.

Evaluating User Interfaces with Metaphors of Human Thinking

Kasper Hornbæk[1] & Erik Frøkjær[2,*]

[1] Natural Sciences ITC Competence Centre, University of Copenhagen,
Universitetsparken 5, DK-2100 Copenhagen,
KHornbaek@nik.ku.dk
[2] Department of Computing, University of Copenhagen,
Universitetsparken 1, DK-2100 Copenhagen
erikf@diku.dk

Abstract. Inspection techniques are a useful tool for identifying potential usability problems and for integrating at an early stage evaluation with design processes. Most inspection techniques, however, do not consider users' thinking and may only be used for a limited range of devices and use contexts. We present an inspection technique based on five metaphors of essential aspects of human thinking. The aspects considered are habit; the stream of thought; awareness and associations; the relation between utterances and thought; and knowing. The proposed inspection technique makes users' thinking the centre of evaluation and is readily applicable to new devices and non-traditional use contexts. Initial experience with the technique suggests that it is usable in discussing and evaluating user interfaces.

1 Introduction

This paper presents an inspection technique for evaluating user interfaces. The technique is based on five metaphors of human thinking. The technique aims at helping evaluators consider users' thinking when evaluating user interfaces.

Inspection techniques for evaluating user interfaces aim at uncovering potential usability problems by having evaluators inspect the user interface with a set of guidelines or questions [29]. Inspection techniques seem to be a useful supplement to empirical techniques for identifying potential usability problems and for integrating at an early stage evaluation with design processes. Inspection techniques include heuristic evaluation [30], where inspection is guided by heuristics such as "Be consistent" or "Prevent errors" [30], p. 249; and cognitive walkthrough [21,42], where evaluators ask questions related to how users perceive the user interface and plan actions.

To us, existing inspection techniques have two shortcomings. The first is that most inspection techniques do not explicitly consider users' thinking. Guidelines, for example, often do not mention users' thinking or relate to psychological principles. The first 37 guidelines in the collection by Smith & Mosier [39] refer to users' thinking or psychological principles in only 10 cases: however, this is done in superficial phrases

[*] The authors contributed equally to the paper.

such as “Most users will forget to do it” or “People cannot be relied upon to pay careful attention to such details” [39], p. 34. Heuristic evaluation, as in [30], mentions only the user explicitly in two heuristics, and the heuristic “minimize users’ memory load” is the only heuristic that comes close to considering users’ thinking. One exception to our point, it may be argued, is cognitive walkthrough, which was developed with a basis in psychological theories of exploratory learning [21]. However, even for this inspection technique refinement has led to less emphasis on the psychological basis. In [42], the original list of nine questions (some with sub-questions) was reduced to four by some of the inventors of cognitive walkthrough. And recently, in the so-called stream-lined cognitive walkthrough [41], only two questions are asked with no 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?”, [41] p. 355. So existing inspection techniques consider users’ thinking only vaguely, thus ignoring potential important insight in how thinking shapes interaction.

A second shortcoming of inspection techniques is that many of the guidelines or questions used are useful only for a particular device/interaction style (e.g. Windows Icons Menus Pointers-interfaces) or context of use (e.g. computing in front of the desktop). Again take Smith & Mosier [39] as an example: their guidelines discuss the use of “reverse video”, “the ENTER key” and “Teletype displays”—all of these obviously is linked to a certain device/interaction style. A related concern was recently stressed by Preece et al. who in a discussion of heuristic evaluation writes:

> “However, some of these core heuristics are too general for evaluating new products coming onto the market and there is a strong need for heuristics that are more closely tailored to specific products.”, [35], p. 409

It is not clear, therefore, that mainstream inspection techniques can be successfully used to inspect emerging technologies, such as mobile devices or context aware systems. Pinelle and Gutwin [32] found too little focus on the work context in inspection techniques used on groupware and have tried to extend cognitive walkthrough to include such a focus. This is just one example of the limited transferability of inspection techniques to other use contexts.

As a solution to these problems, and as an effective inspection technique in its own right, we propose an inspection technique based on Metaphors Of Thinking—the MOT-technique. We use the term thinking broadly to denote mental activity as this takes place for a person as part of his or her activities. The MOT-technique is inspired by William James's and Peter Naur's descriptions of human thinking [19,23-26]. Similar descriptions along with many brilliant design discussions have lately been introduced to HCI in Jef Raskin's book *The Humane Interface* [36]. Naur's and Raskin's work are complementary to most psychology used in HCI, but is supported by extensive evidence from classic introspective psychology [19], and from experimental psychology and neurology [1,2]. Several of the aspects of human thinking described in this work are of critical importance to successful human-computer interaction: (1) the role of habit in most of our thought activity and behaviour—physical habits, automaticity, all linguistic activity, habits of reasoning; (2) the human experience of a stream of thought—the continuity of our thinking, the richness and wholeness of a person's mental objects, the dynamics of thought; (3) our awareness—shaped through a focus of attention, the fringes of mental objects, association, and reasoning; (4) the incompleteness of utterances in relation to the thinking underlying them and the ephemeral nature of those utterances; and (5) knowing—human knowing is always under construction and incomplete.

We base the MOT-technique on these five aspects that combined and separately catch important properties of thinking shared by humans. Each aspect of thinking is described also by a metaphor. The metaphor is meant to help the evaluator in understanding and focusing on the aspect considered; it is *not* intended as an interface metaphor. The number of metaphors is not important; nor is our choice of metaphors meant to catch all aspects of human thinking. Still we hope to show that the metaphors are valuable in evaluating user interfaces.

We have chosen to present these aspects of human thinking by quotations from James [19] and Naur [25-27]. Naur has studied the 1377 pages of James's book *The Principles of Psychology* and through quotations, summaries and extended discussions illuminated James's work and to us made it more accessible. For readers who might not be aware of the continued importance of James's classical work in psychology, and who therefore might feel uncomfortable with our paper's building so directly on sources published more than hundred years ago, we quote the renowned cognitive psychologist Bernard Baars who in 1997 writes:

> 'Remarkably, the best source on the psychology of consciousness is still William James's elegant 'Principles of Psychology', first published in 1890. [...] James' thoughts must be understood in historical context, but the phenomena he describes so well have not changed one bit.' [2], p. 35

Table 1 summarizes the metaphors, indicate the implications for user interfaces, and give examples of key questions that the metaphors raise. The following five sections (section 2 to section 6) describe in detail each metaphor. In section 7 we discuss how to conduct an inspection using the MOT-technique, and section 8 discuss the potential and problems of the MOT-technique.

2 Support of Existing Habits and, when necessary, Development of New Ones

Fundamental for effective use of interfaces is that users can use existing habits, such as common association of a word to an object or using certain key combinations for inserting text. Similarly, the user should be able to develop effective work habits with the interface: it should be possible to predict layout and functioning of the interface, accelerators should be provided, it should be easy to accomplish frequent work tasks, etc.

2.1 Habit as a Landscape Eroded by Water

Every person's habit formation is like a landscape eroded by water. By this metaphor we mean to indicate how a person's formation of habits leads to more efficient actions and less conscious effort, like 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.

Table 1. Summary of the MOT-technique. The five metaphors, their implications for user interfaces, and examples of questions to be raised

Metaphor of human thinking	Implications for user interfaces	Key questions/Examples
Habit formation is like a landscape eroded by water.	Support of existing habits and, when necessary, development of new ones.	Are existing habits supported? Can effective new habits be developed? Is the interface predictable?
Thinking as a stream of thought.	Users' thinking should be supported by recognizability, stability and continuity.	Do the system make visible and easily accessible the important task objects and actions? Do the user interface make the system transparent or is attention drawn to non-task related information? Does the system help users to resume interrupted tasks? Is the appearance and content of the system similar to the situation when it was last used?
Awareness as a jumping octopus.	Support users' associations with effective 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? Is the graphical layout and organization helping the user to group tasks?
Utterances as splashes over water.	Support changing and incomplete utterances.	Are alternative ways of expressing the same information available? Are system interpretations of user input made clear? Do the system make a wider interpretation of user input than the user intends or is aware of?
Knowing as a site of buildings.	Users should not have to rely on complete or accurate knowledge—design for incompleteness.	Can the system be used without knowing every detail of it? Do more complex tasks build on the knowledge users have acquired from simpler tasks? Are feedback given to ensure correct interpretations?

According to James the most important general property of the thinking and behavior of people is that each person is a bundle of habits. Building on James, Naur writes [27]:

> 'All our grasping of things around us that we see, hear, feel, that which we call perception, is entirely a question of the habits each of us has trained. In addition our locomotion, the way we move our arms and legs while moving around, is almost entirely habitual. In addition, our talking with each other, the way we grasp what others say to us and the way we move our tongue, lips, and other organs of speech while talking, all this has been trained as habits. All education is a matter of training habits.
>
> Any part of a human organism may be involved in a habit. In a certain sense every habit involves the entire person.'

2.2 Key Questions

The key questions for already established habits are whether they are transferable to the user interface in question. Can often-used shortcuts or common associations between command names and functions known from other applications be used? Are actions that are executed almost automatically by many users supported, e.g. pressing return after entering a query word? On the other hand, be cautious in the use of modes as they may hinder transfer of habits.

A key question for users is whether the user interface allows for habit formation of often-used actions. Can the central user tasks be effectively done in the user interface? To allow users develop new habits, the interface should be predictable and responsive. When an interface is predictable, information and controls (e.g. menus), appear in the same place every time the program is used. Such predictability allows users to begin moving the mouse towards a menu item or a button even before they have oriented themselves in the interface. Responsiveness in this context means that the interface should allow the user to begin typing commands or pressing buttons immediately, for example even when parts of a web-page is still loading. Similarly, the user should be allowed to type ahead in menus or forms.

2.3 Examples

There is an abundance of examples of user interfaces that violate human habits. One example is adaptive menus, used for example in Microsoft Office 2000, see Fig. 1. 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 seldomly used. However, adaptive menus make it impossible to form habits in the selection of menu items [36], since their position may be different from when they were previously selected. A study by Somberg [40] 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. Note that the common practice of adding a fixed number of, say, recently used files or fonts to the bottom or top of a menu does not interfere with habit formation and may decrease time taken to select a menu item [37].

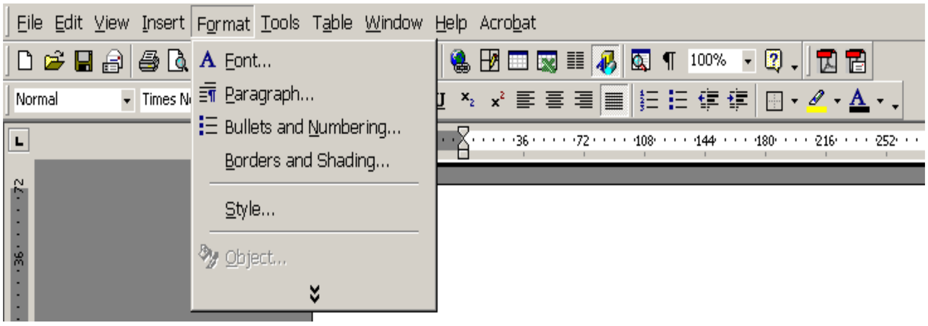

Fig. 1. Adaptive menus in Microsoft Word. Some menu items in the Format menu have been hidden based on the frequency with which they have been used. This, however, prevents the user from forming habits

The discussion of consistency in user interfaces may be illuminated in terms of habit. In a classic paper on consistency [13], Grudin argues that focusing on consistency per se leads to a lack of focus on users and their tasks. In several examples he shows how consistency can be interpreted in different ways and how different aspects of usability contradict each other in what some call consistent designs. From our point of view, Grudin's critique of the notion of consistency concerns the role of habit in the interface. With a focus on habits, the aim of consistency is to allow the habits that users develop to be transferred within or between systems they use. In addition, a system should also allow effective habits to be established in the first place, especially for often-used functions. Consistency between systems is not critical if interface elements or functions are not a habitual part of the users' repertoire of actions. Habitual association of words, however, might be useful for grouping or naming interface elements.

The central design issue with respect to consistency, and thus habit formation, is whether to utilize existing habits in the design of the system or create new ones. Grudin's [13] discussion of choosing effective keyboard layouts (e.g. QWERTY or DVORAK) is an example where it is essential for users to establish effective new habits, rather than transferring real-world habits (such as associating letters in alphabetical order) to the interface. One reason why consistency is a problematic notion is that it obscures long-term usability—especially the efficiency gained by supporting inattentive, i.e. habitual, use.

Perhaps designers in HCI more often should aim for establishing new, effective habits. Even the most radical changes of interfaces may be mastered if the interface is used often. An analogue of this is shown in Stratton's experiments with glasses that turned his visual field upside down [12]. When wearing the glasses constantly, in less than 7 days he had become habituated to viewing the world upside down and could walk, write, etc.

An example of a user interface that exploits that habit formation is not always wanted, is found in the evaluation version of the compression utility WinZip, see Fig. 2. When WinZip is run, an initial screen with five buttons is shown. Three buttons allow the user to get access to license information, to a screen for registration, and to information about how to order. The last two buttons are of interest here. One button quits the utility; another lets the user proceed to the main screen of WinZip. To prevent users from going straight to the main screen, the designers of WinZip randomly interchange the position of the two buttons when the utility is run. This prevents the user from establishing a habit of clicking the proceed button without noticing the license and ordering information on the initial screen.

As pointed out by Raskin, many error messages will not be noted by the user if they often or always appear when something potentially harmful are initiated:

> ‘The inevitability of habit formation has implications for interface design. For example, many of us have used computer systems that, before they will perform an irreversible act, such as deleting a file, ask, ‘Are you sure?’ You then must type, say, a *Y* for yes or an *N* for no in response to the question. This idea is that, by making you confirm your decision, the system will give you a chance to correct an otherwise irrecoverable error. This idea is widely accepted. For example, Smith and Duell (1992), addressing a nursing environment, say, ‘If you inadvertently delete part of the permanent record (which is hard to do because the computer always asks if you’re sure)…’ (p. 86). Unfortunately, Smith and Duell are unrealistic in their assessment: you can readily make an incidental deletion even when this kind of confirmation is required. Because errors are relatively rare, you will usually type *Y* after giving any command that requires confirmation. Due to the continual repetition of the action, typing *Y* after deleting soon becomes habitual. Instead of being a separate mental operation, typing the *Y* becomes part of the delete-file action; that is, you do not pause, check your intentions, and then type the *Y*. The computer system’s query, intended to serve as a safety measure is rendered useless by habituation; it serves only to complicate the normal file-deletion process. The key idea is that *any confirmation step that elicits a fixed response soon becomes useless.* Designers who use such confirmations and administrators who think that the confirmations confer protection are unaware of the powerful habit-forming property of the cognitive unconscious.’ [36], p. 22

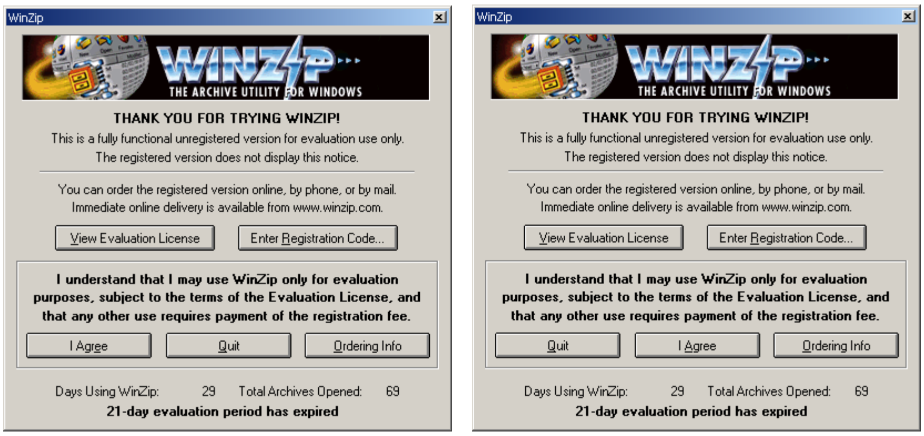

Fig. 2. Two versions of the initial screen in Winzip, which shows how the position of the ‘I Agree’ button change, so as to ensure that users pay a minimum amount of attention to the information on the screen

3 Users' Thinking Should be Supported by Recognizability, Stability and Continuity

Mental activity is most effective when not interrupted and may be concentrated and flowing when the activity is challenging, yet well known. Mental activity is constantly changing as a result of inner and outer factors, for example physiological, psychological, and social—and of situation-specific factors: what has happened for the person up to now and what is the expectations for the near future. Thus mental activity is a very complex unity. A user interface should complement this changing complexity with recognizability, stability and continuity.

3.1 Thinking as a Stream of Thought

The metaphor of human thinking as a stream of thought is the result of James's own choice. He says [19], vol. I p. 239:

> 'Consciousness, then, does not appear to itself chopped up in bits. Such words as 'chain' or 'train' do not describe it fitly as it presents itself in the first instance. It is nothing jointed; it flows. A 'river' or a 'stream' are the metaphors by which it is most naturally described. In talking of it hereafter, let us call it the stream of thought, of consciousness, or of subjective life.'

Naur summarizes James's description of human thinking as a stream of thought in this way [26], p. 85:

> 'In William James's Principles of Psychology the stream of thought denotes something happening in all of our wake moments, to wit our experience of thinking and feeling. The stream of thought is known to every one of us through introspection, that is through our turning the attention inward, towards the way we experience our thoughts and feelings. What we may register through introspection is merely a picture of rough outlines. The stream of thought changes incessantly and has a vast number of details, most of which are present only vaguely, far more than may be seized by introspection.
>
> The stream of thought happens independently of our desire. We may, when we so wish, more or less successfully think of something definite, but we cannot make the stream of thought cease, as experienced by every person suffering from insomnia.
>
> The stream of thought may be described as something that flows, an incessantly changing, complicated mixture of something that may be denoted explicitly as images, sounds and bodily impressions, with additional vague moods and feelings. As stressed by James we do not in the stream of thought experience sharply delimited parts or elements of any kind. At each moment our thought is occupied by something that is complicated, but that is experienced as a whole. These wholes James calls thought objects [Our remark: also called 'mental objects']. Within each thought object one may distinguish between something more at the center, that which is the subject of our attention, and something that forms a fringe. […] [E]very thought object embraces feelings, including those of the personal well-being, moods and bodily presence.
>
> In its continued changing the stream of thought alternates between substantive states of relative repose and transitive states of rapid change. During the transitive states the changes of the thought objects happen so rapidly that they cannot be seized by introspection.
>
> In the experience of the stream of thought the present moment has a duration of a few seconds. As one thought object fades away by being replaced by another one, it is retained in

> the fringe of the coming one. Every sudden impression is always experienced as a whole with what was there immediately before it happened.'

3.2 Key Questions

User interfaces should respect these traits of mental activity. The objects and actions that the user focus on when doing tasks should be clearly visible in the user interface. Users' thinking about their tasks is centred on a few task objects. Does the application make these task objects visible and easily accessible? Is the user interface of the application transparent, meaning that it does not demand attention on non-task related objects and actions? Recognizability of objects is a key priority; as is the relevance of user interface objects and actions to the users' tasks.

The description of the stream of thought makes it clear that interruptions, breaks, and pauses are characteristic of human mental activity. Another key question is how well the application supports the interruption and resumption of work. Most work done with computers is interrupted or discontinued for seconds, hours, or months. Does the application help users come back to their tasks, keeping available the situation from before the interruption or discontinuation? Does the user interface help the user establish part of the stream of thought experienced when the work was interrupted? Are the appearance and content similar to the previous occasion on which the program was used? Can the user resume interrupted or old work easily? Does the user have direct access to previously used files and directories?

Another key question concerns the changing of thought and the influence of an application on changes in our thoughts. Error messages, e-mail notifications or information on windows updates are likely to change users' stream of thought. Such messages should be used sparingly. System initiated dialogs are in principle harmful, when not about critical failures, and should be kept at a minimum. Often used functions should be easily locatable and have default values, so as to disturb the users' stream of thought as little as possible.

Another way of supporting the users' changing stream of thought is to provide stability and predictability in the interface. Key questions are whether changes in the interface are kept to a minimum and whether actions have stable and predictable results.

3.3 Examples

One example of poor support for the resuming of an interrupted task is found in the application SPSS (see www.spss.com), intended for statistical analyses. In SPSS, it is possible to specify certain statistical analyses with syntax. Such analyses are always run on a data file. However, SPSS does not recreate or save information about which data file a syntax file is associated with. So, when one loads a syntax file and tries to run it, error messages occur, and one has to figure out which file, possibly out of a number of files, you have made the syntax file for. A better solution would be to automatically run the syntax file on the data file it was created with or last executed.

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 design of the Canon Cat [36]. 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. It also provides help in remembering and recreating the stream of thought as it was when work was interrupted.

The fragility of the stream of thought is not well protected in many user interfaces. E-mail notifications, instant messengers, news on demand, automatic spelling and grammar corrections are useful at times, but may also disrupt concentrated work. Research on instant messengers, for example, has documented the harmful effects of interruptions on task completion time [8]. As a personal note, one of the authors of this paper has recently removed all notifications of arriving e-mails from his computer. Even the .5 cm ×.5 cm icon in the lower right corner of the screen that show the arrival of new e-mail could create an intense feeling of urge to check the e-mail—which would initially be in the fringe of the current object of attention, but eventually would lead the author to start of the e-mail program. This seemed especially to happen when that author was struggling with a difficult task. In general, we find that most user interfaces fail to support shifting between what we experience as two phases of work: concentrated working, where interruptions and distractions are detrimental, and explorative working, where a free flow of associations, inspirations, breaks, and even interruptions can be useful.

An example of the dynamics of thinking that is closely related to the stream of thought is found in information retrieval studies concerning changes in relevance judgments of documents. One study [9] showed that the order in which subjects viewed document descriptions influenced the subjects' perception of the relevance of those descriptions. While this effect in part may be due to the categorical rating scales used, a psychological explanation is also possible. When looking at document descriptions, the themes of the previous descriptions will be in the fringe of the subject's mental object. Those fringes will influence the perception of the task and the judgment of the current document description. Thus, different orderings of documents will give different relevance judgments; significant differences in relevance judgments were found even between random orderings of the documents to be judged. Thus, relevance judgments seem to be dynamic in a sense closely related to the metaphor of the stream of thought.

An often-added interface feature that helps coming back to an interrupted task is a list of recently used files (see Fig. 3), even though such file names are often heavily truncated. Such information often makes sense to users. A similar idea is to save the entire layout of an application, so that the application will look similar next time it is run.

The recognizability of central objects and actions in the users' task are recommended in many methods and tools for systems development. One example is Ben Shneiderman's Object-Action Interface Model, [38] p. 61, which suggests that objects and actions from the task domain should form the basis of the objects and actions in the interface. Such recommendations are coherent with the description of thinking as a stream of thought in which objects that the user is acquainted with are anchored.

4 Support Users’ Associations with Effective Means of Focusing within a Stable Context

4.1 Awareness as a Jumping Octopus

‘The mental activity is like a jumping octopus in a pile of rags’, says Naur [25] and continues to illustrate the dynamics of thinking:

> ‘This metaphor is meant to indicate the way in which the state of consciousness at any moment has a field of central awareness, that part of the rag pile in which the body of the octopus is located. The arms of the octopus stretch out into other parts of the rag pile, those parts presenting themselves vaguely, as the fringes of the central field. […] The jumping about of the octopus indicates how the state of consciousness changes from one moment to the next.’

The rags of the pile may through focusing come to the field of central awareness. Here associations play a central role. On this Naur [26], p. 11, summarizes from James:

> ‘One object of thought is replaced habitually by the next. We say then that the two thoughts are associated or that the next thought appears through its association to the first one. [...] [W]hat enters into the association of thoughts is not elementary 'ideas', but complicated *thought objects* which are experienced as wholes but each of which includes more central parts and a *fringe* of vague connections and *feelings*.’

Associations may happen by contiguity and by similarity. Association by contiguity is essentially a matter of habit formation. James [19], vol. I, p. 561 says:

> ‘[...] objects once experienced together tend to become associated in the imagination, so that when any one of them is thought of, the others are likely to be thought of also, in the same order of sequence or coexistence as before. [...] it expresses merely a phenomenon of mental habit, the most natural way of accounting for it is to conceive it as a result of the laws of habit in the nervous system.’

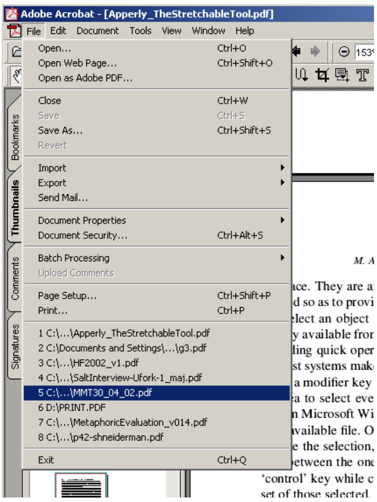

Fig. 3. Support for resuming work. This screen-shot shows the list of recently used files in Adobe Acrobat reader. This way it is efficient for users to come back to recently used files

Association by similarity is:

> '[…] association between thought objects that have become connected in the thought merely by having the same abstract property in common, in other words by being similar in some respect.' [26], p. 12

Association by similarity plays an important role in reasoning. Reasoning is concerned with solving problems, or answering questions, related to situations involving certain known things, having certain known properties, in which the person cannot reach the solution or the answer by direct association from the known properties. James explains how successful reasoning builds upon the person's noticing and attending to certain definite properties of the situation at hand, to wit such properties that point to a way of reaching the goal by direct association. James makes clear how reasoning in this sense is a decisive factor in human inventiveness and discovery, including that of scholars and scientists, see [27].

4.2 Key Questions

One key question concerns which associations users create in response to user interface objects. Such associations can happen by physical proximity, habitually associations of words, relation to central task objects, etc. Do users associate interface elements with the actions and objects they represent? Can words and labels in the interface be expected to create useful associations for the user? Are contiguity and similarity used in the graphical layout to help the user to group and understand tasks?

User interfaces should help users keep focus and should respect associations that make users wanting to switch between different parts of the interface. Key questions are: Does the user interface force the user into a specific order of tasks? Are modal dialog boxes used more than necessary? Can the user switch flexibly between different parts of the interface?

In general, our thinking when using computers has a main focus and an associated host of feelings, vague connections, associations, etc. The main focus will often correspond to part of the user interface. User interfaces should respect this. Do not expect the user to focus only on specific parts of the user interface. For example, when users go back and forth between web-pages in order to correct information, they are not likely to notice that information is deleted or changed.

4.3 Examples

Modal dialog boxes prevent 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 began.

The metaphor of the octopus can be illuminated by recent studies of awareness, e.g. [11,14]. Common to these studies is an aspiration to design for peripheral awareness, to design also for the fringes of the octopus so to speak. As an example consider Grudin's study of how multiple monitors are used [14]. Grudin found that among 18 users who used multiple monitors simultaneously, the multiple monitors were not used as additional space, but to partition the information used. Users would for example delegate secondary tasks such as debugging windows in a programming environment to the second monitor, and some users would have e-mail, news alerts, and instant messengers on the secondary monitor. Grudin's study is coherent with and supportive of the metaphor of awareness in two important ways. First, users employ the degree of attention they give information to dived their work between monitors. Less important information is in the periphery of the eye and thereby to some extent in the fringes of the current mental object. This may reflect how users introspectively realize that some information sources may in subtle ways distract us, but that they may be useful for creating fringes. Second, Grudin's work and other recent papers on awareness show opportunities for designing for peripheral attention, and even for inattentive use of computers. It is evident from the metaphor of the octopus that the fringes of mental objects form a large part of our thinking—and this should be taken into account when designing.

The characteristics of awareness and the association of objects thought of with other objects are not unfamiliar descriptions of human thought in HCI. Already Vannevar Bush in his vision of the Memex [4] exemplify this:

> 'When data of any sort are placed in storage, they are filed alphabetically or numerically, and information is found (when it is) by tracing it down from subclass to subclass. It can be in only one place, unless duplicates are used; one has to have rules as to which path will locate it, and the rules are cumbersome. Having found one item, moreover, one has to emerge from the system and re-enter on a new path.
>
> The human mind does not work that way. It operates by association. With one item in its grasp, it snaps instantly to the next that is suggested by the association of thoughts, in accordance with some intricate web of trails carried by the cells of the brain. It has other characteristics, of course; trails that are not frequently followed are prone to fade, items are not fully permanent, memory is transitory. Yet the speed of action, the intricacy of trails, the detail of mental pictures, is awe-inspiring beyond all else in nature. Man cannot hope fully to duplicate this mental process artificially, but he certainly ought to be able to learn from it.'

However, as pointed out by Wendy Hall at the Hypertext'01 Conference, links that take the user to web pages associated with the link description are fairly uncommon at the web [15]. In hypertext research, such links are called associative or referential links [7], as opposed to for example navigational or organizational links. According to Hall, less than 1% of links on the World Wide Web are associative: the rest are predominantly navigational links. On one side this suggests that Bush's warning has been taken seriously—human awareness and association are not directly modelled on the WWW. On the other side, we feel that the lack of associative links might suggest that designers have paid too little attention to awareness, associations, and how to craft links that use this fundamental trait of human thinking.

As an example of a notion in HCI that may become clearer from the metaphor of the octopus, we would like to briefly discuss information scent. Information scent refers to:

> '… the (imperfect) perception of the value, cost, or access path of information sources obtained from proximal cues, such as bibliographic citations, WWW links, or icons representing the sources' [33]

In HCI this notion has recently received much attention in relation to web design [5]. From our perspective, information scent is the ability of proximal cues to create in the mind of the user associations related to the content looked for. The degree to which WWW links or icons have 'information scent' is only a matter of the associations they create for individual users. In some studies of information scent, e.g. [34], an information scent score is developed. Subjects are given the top levels of a hierarchical link structure and the information scent score is the proportion of subjects who correctly identify that a certain link contains the answer to some task. Thus, subjects assess the links from the associations created in relation to the task. The second aspect of the definition of information scent—the cost of accessing information sources—is related to habit. We most often follow our habits in traversing information structures rather than pondering the cost of certain ways of navigation. Information scent may in this way be easily understood through the metaphors of awareness and habit.

The terminology used in a computer application is sometimes technical and obscure; for some users such terminology will if they read it fail to create useful associations.

5 Support Changing and Incomplete Utterances

Human utterances are often vaguely related to the intentions behind the utterances. In addition, humans use a variety of ways for expressing similar intentions. Applications should be designed so as to respect this changing and incomplete character of all utterances, whether they are mouse movement, voice input, typing, gestures or other means of input. Special care should be taken when applications use utterances to infer the insights and feelings behind them.

5.1 Utterances as Splashes over Water

‘A person's utterances relate to the person's insights as the splashes over the waves to the rolling sea below’, says Naur [25] and continues:

> ‘This metaphor is meant to indicate the ephemeral character of our verbal utterances, their being formed, not as a copy of insight already in verbal form, but as a result of an activity of formulation taking place at the moment of utterance.’

The metaphor also emphasizes how utterances are vague and incomplete expressions of the complexity of a person’s current mental object, in the same way as the splashes tell little about the sea below.

5.2 Key Questions

Applications should accommodate the variety and flexibility in which humans express themselves. For example, are different ways of expressing the same information (such as centimeters, inches, points, etc. as input for length) available? If file names, logins, naming, short-cuts, or other user utterances are part of the user interface, users should be supported in recalling which utterance they last used. If a fixed format or content of an utterance for some reason is needed, the application should help the user as far as possible. For example, if calendar dates should be input in a special format, that format should be exemplified and fields coded to suggest the order in which to input days and months, and the number of digits required when entering a year.

Another key question concerns how the application interprets users’ utterances. What inferences about user intention and insight are drawn, and how do these affect the interaction? Does the application make a wider interpretation of user input than warranted? Are interpretations made clear and are they easily understandable? Is feedback given about what the system is doing as a result of a user utterance? The last point is especially important for utterances that require interpretation on the part of the system.

5.3 Examples

One implication of the metaphor of utterances as splashes over the water is that we must expect users to describe the same objects and functions incompletely and in a variety of ways. Furnas et al. [10] investigated the diversity in words used for describing commands and everyday objects. On the average, two participants described the same command or object by the same term with less than 20% probability. The most popular name was chosen only in 15-35% of the cases. Furnas et al.'s suggestion for relieving this problem is called the unlimited alias approach. Instead of using a fixed set of words for commands and functions, the unlimited alias approach lets users enter any term they want. If the term is not in the range of terms initially suggested by the designer of the system—which the data of Furnas et al. and the metaphor suggest it often will not be—the system may interactively suggest appropriate commands or object names. This approach is coherent with the metaphor and uses interactivity to clarify the intentions of the user. On the other hand, the approach partly goes against the metaphor of habit formation.

We believe that the relation between queries made on the WWW and what users are looking for may be made easier understandable by use of the metaphor. Queries on the WWW are on the average 2.2 words long [20]. However, such short queries cannot possibly reflect all aspects of the pages users are looking for, nor can they reflect the myriads of interests, questions, etc. that may suddenly become the locus of attention when triggered by otherwise irrelevant web pages. In information retrieval, the difficulty in interpreting the intention (or information need) behind the queries has long been recognized as problematic, as have the difficulty of expressing one's information need in the first place [3]. Harter [16] has gone so far as to suggest that the information need is indeed our full mental constitution—which is impossible to express in a few words or queries. This is in accordance with the metaphor of utterances as splashes over the water and respects the complexity of mental objects, as described by the stream of thought and the octopus metaphors.

There are numerous examples of user interfaces that do not respect the metaphor of utterances. Many of these involve systems that try to predict, given a few utterances, the needs and wishes of the user—something that is unlikely to succeed given the ephemeral and incomplete nature of utterances. One example is the attempt of the Office Assistant in Microsoft Word to infer which kind of document the user is writing given one or two words from that document. Another is the annoyance of screen savers or sleep modes of laptops, if they are initiated by interpreting the lack of user input as a lack of user work with the application.

If the application interprets users' utterances, then it should make the interpretation clear. This helps the user understand subsequent output. One simple example of this is the use of feedback when searching at Google, see Fig. 4. When the users' input is interpreted, and in this case partly ignored, an explanation is given.

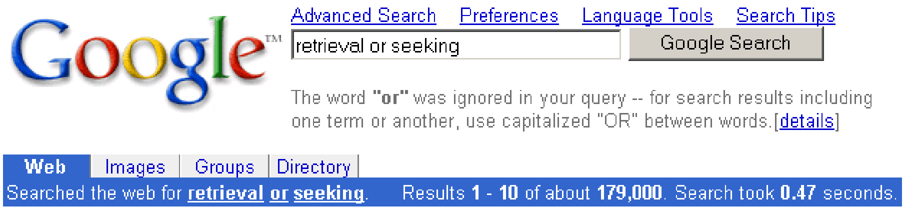

Fig. 4. Example of feedback when interpreting user input. When using the term ‘or’ in a search, the Google website (www.google.com) will give an explanation of the interpretation made of that term

6 Users Should not Have to Rely on Complete or Accurate Knowledge—Design for Incompleteness

Human knowing, for example of tasks and user interfaces, is constantly changing. In addition, much of what we know is incomplete, inconsistent, and even seriously flawed. Applications should respect these traits of human knowing and take them into account as far as possible. Conversely, users should not have to rely on complete or accurate knowledge about applications.

6.1 Human Knowing as a Site of Buildings

Human knowing is like a site of buildings in an incomplete state of construction, developed through maintenance and rebuilding. In Naur's [25] formulation:

> ‘A person's insight is like a site of buildings in incomplete state of construction. This metaphor is 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 employed as shelter, so the insights had by a person in any particular field may be useful even if restricted in scope. And as the unfinished buildings of a site may conform to no plan, so a person may go through life having incoherent insights.’

6.2 Key Questions

One key questions concerns whether both novices and experts can work effectively with the application, given that their knowing about the application and task domain may be very different. This concerns also to what degree applications support users in developing an understanding of that application. In general, users can only be expected to develop knowing about an application to at most the level that will enable them to complete the task. Users will therefore have insecure and shaky knowledge about a range of applications.

Other key questions are the following. Can users start using the application immediately or is it required that the user pay attention to technical or configuration details? Can the application be used without knowing every detail of it? Can simple tasks be completed in a simple way? Do more complex tasks build on the knowledge users may have from simple tasks? The last two questions concern whether all users can effectively accomplish simple tasks and whether effective habits (see section 2) may be developed.

Another aspect of human knowing is that it is often incomplete. This is especially true of applications that are seldomly used, and of knowing about the internal workings of hardware and software. Applications should respect this trait of human knowing. Key questions concerning this aspect are as follows. Are the users supported in remembering and understanding information and relations in the application? Is feedback given to ensure correct interpretations? Are error situations handled in a graceful way that supports the users' possibly limited understanding of the error?

6.3 Examples

Similar to the advice of designing for incomplete knowing is the HCI maxim "support recognition over recall" [38]. One example of this principle in use is given in Fig. 5.

Examples where the metaphor of a person's knowing is not respected are easy to find. Systems that require a full understanding of the system before they may be used are cases in point. An example is described in Chen & Dhar's study [6] of an online library catalogue. They observe how 30 subjects take wrong actions in using the system, how they use wrong query terms, and how they use a sub-optimal procedure for accomplishing tasks. The faulty actions arise from the subjects' misconceptions about the topic they are searching for, about the way the online catalogue works, and about the nature of the classification system used. Each subject displayed at least one misconception. First of all this shows that even for a common task like searching a library system, the subjects' knowing about the program was incomplete. Second, Chen &

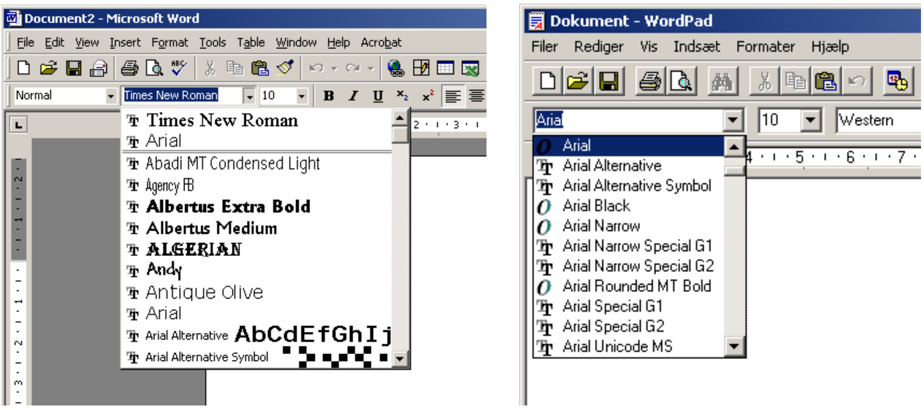

Fig. 5. Two ways of specifying fonts. The left screen shot shows Word, which succeeds in supporting an incomplete knowing of the appearance of various fonts. The right screen shot shows WordPad, which requires the user to know the appearance of fonts or to try them out

Dhar's results show that the design of the online catalogue violated the metaphor of the site of buildings in several ways. As one example, the system only recognizes official Library of Congress subject headings, which in essence requires the subjects to have a complete and precise understanding of how their problem relate to the official terms. The lack of support for cross-referencing and inferring correct headings worsens this.

Another example of ignorance of the idea of the developing and incomplete nature of human knowing, is the use of technical information in a number of applications that are to be used by users who do not know the technical details of how a program works. A number of applications seriously flawed in this way is shown on http://www.iarchitect.com/, Fig. 6 shows one example accompanied by the following explanation:

> "*Microsoft Word 6.0* when asked to open a document from an unknown version of *Word* displays the above message. *Word*'s conversion utility seems to be asking, "I think it's a Word '97 document, but it might be one of these other types. What do you think?" **How the #$%@& would I know!** This is the result of a confirmation-happy programmer. The conversion utility knows exactly what type of file it is (the raw file contains two explicit references to the type of file), yet the program wants the *user* to confirm the program's ability to read these references. The user, on the other hand, unless he or she created the file, has absolutely no knowledge of the file type. By needlessly asking the user, the program needlessly creates uncertainty and an opportunity for the user to cause an error."

Mental models have been extensively discussed in HCI. Consider as an example Norman's [31] description of the use of calculators. He argues that the use of calculators is characterized by users' incomplete understanding of the 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 empirical observations by Norman are coherent with the ideas expressed by the metaphor of knowing. In summary, the library catalogue and the use of calculators show that users solve the actual tasks despite inconsistencies and incompleteness of their knowing. Conversely, systems that require a precise and complete understanding are often awkward to use.

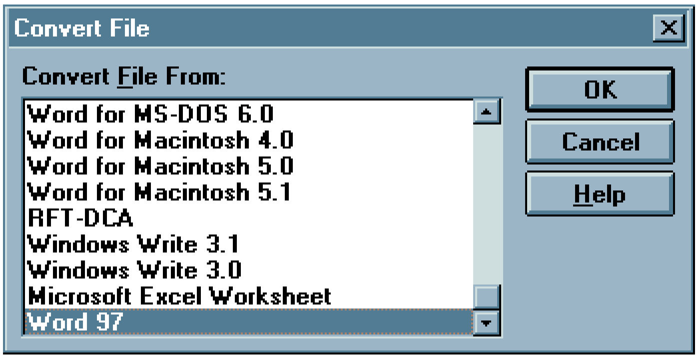

Fig. 6. Dialog asking the user to specify the program a document was created with. One problem with this dialog is that it assumes that user's technical knowledge is complete and that users actually know with which system the file was created

7 How to Do a Metaphor-based 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 captured in the metaphors and the key questions. This enables the evaluators to identify potential usability problems.

More concretely, one way to do an evaluation after having read section 2 to 6 about metaphors of human thinking is to follow the steps below.

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 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 step 3 and 4 for those tasks. Iterate step 3 and 4 until each new task reveals few new problems or until no time is left.

The above procedure is normally carried out by one evaluator. Several studies have documented that different evaluators find different problems [18,28]. Therefore, it may be useful to combine lists of problems with those found by other evaluators who have also inspected the application using the metaphors. The evaluations must be carried out individually before results are discussed and combined.

The duration of an evaluation based on the metaphors will normally be between one and two hours.

The evaluation will result in a list of usability problems, each described with reference to the application and to the metaphor that was 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.

We consider it likely that different evaluation methods will find different usability problems and give different feedback about the nature of the problems. Evaluators are well advised to combine the evaluation based on metaphors with other evaluation methods, such as think aloud [17,22] or heuristic evaluation [28,29].

8 Discussion and Conclusion

This proposal of a new inspection technique must be critically evaluated from at least the following points of view:

1. Can the MOT-technique be used effectively and efficiently in revealing important usability problems of new designs? And if so—what kind of training is necessary for the evaluators? In different stages of system development, how should the evaluation process be organized? Does the technique work better if adjusted or

supplemented for specific contexts of usage, e.g. types of devices, interaction styles, or types of users?
2. Do the evaluators consider the study and training of the MOT-technique to be relevant, easy and interesting? Are the five metaphors well chosen, individually and combined?
3. After acquisition, how do the evaluators and system developers use the technique in their design work—is the MOT-technique convenient and adequate and how is it combined with other design and evaluation techniques?

We are far from being able to answer these questions. Many empirical studies involving other researchers, system developers, and users of IT-based devices need to be completed. However, a few initial answers can be reported.

Ad 1: In a major HCI evaluation experiment, designed as project work for 87 computing science bachelor students, a medium complex web application was investigated for usability problems using the MOT-technique and the heuristic evaluation technique [28]. The effectiveness of the two techniques showed similar results and to our surprise the students spent a little less time to do the evaluation with the MOT-technique than with heuristic evaluation.

In an advanced course on HCI for computing science master students, all 17 students chose in their final project to use the metaphors of the MOT-technique in their discussions of selected HCI phenomena described in the scientific literature. Their projects covered a wide range of issues such as new interaction devices, Computer Supported Cooperative Work, information visualization, heuristic evaluation, cognitive walkthrough, GOMS, and WIMP interfaces. The projects showed that the students were able to make original and comprehensive use of the metaphors.

Ad 2: The master students mentioned above found the study of the metaphors of the MOT-technique very interesting, but not easy. James's ideas about thinking, especially thinking as a stream of thought and thought objects, did call for a demanding and radical new thinking of ideas that the students had developed over many years, e.g. about consciousness/sub-consciousness, cognitive/mental models, and conceptualization. During the course, the master students seemed to find the five metaphors of the MOT-technique well chosen and useful in the discussions of psychological aspects of HCI phenomena. Only the jumping octopus metaphor of awareness did cause some 3-4 students a little trouble, especially before it was used concretely in discussions. These students found the picture of the jumping octopus in a pile of rags to be too unrealistic as a "serious" vehicle of expression.

Ad 3: We have not yet carried out any studies that tell us about the usability of the MOT-technique in industry. In the near future we hope to be able to do experiments with system developers and HCI experts. Compared to students, experiments with developers and HCI experts will raise new kinds of challenges, e.g. that we have to "detrain" old habits and ways of thinking before the new ideas in the MOT-technique can be trained. Instead of teaching the metaphors as an inspection technique, it might be more effective to teach developers and HCI experts the metaphors of human thinking as a possible new design vehicle. Afterwards, it can be studied if and how the metaphors are used in design and evaluation activities.

A related question is whether usability problems uncovered by use of the MOT-technique will support system developers better in how to re-design compared to other inspection techniques.

Acknowledgements

We wish to thank Peter Naur for his ongoing sharing of ideas leading us to this paper and his permission to quote extensively from his writings.

References

1. Baars, B. J.: A Cognitive Theory of Consciousness. Cambridge University Press, Cambridge (1988)
2. Baars, B. J.: In the Theater of Consciousness. Oxford University Press, New York (1997)
3. Bates, M.: The Design of Browsing and Berrypicking Techniques for the on-Line Search Interface. Online Review. 13 5 (1989) 407-431
4. Bush, V.: As We May Think. The Atlantic Monthly. July (1945)
5. Card, S., Pirolli, P., Van Der Wege, M., Morrison, J. B., Reeder, R. W., Schradley, P. K., & Boshart, J.: Information Scent as a Driver of Web Behavior Graphs: Results of a Protocol Analysis Method for Web Usability. In: Proceedings of CHI 2001 (2001) 498-505
6. Chen, H. & Dhar, V.: User Misconceptions of Information Retrieval Systems. International Journal of Man-Machine Studies. 32 (1990) 673-692
7. Conklin, J.: Hypertext: An Introduction and Survey. IEEE Computer. Sep (1987) 17-41
8. Cutrell, E., Czerwinski, M., & Horvitz, E.: Notification, Disruption, and Memory: Effects of Messaging Interruptions on Memory and Performance. In: Proc. of Interact 2001 (2001)
9. Eisenberg, M. & Barry, C.: Order Effects: A Study of the Possible Influence of Presentation Order on User Judgments of Document Relevance. Journal of the American Society for Information Science. 39 5 (1988) 293-300
10. Furnas, G. W., Landauer, T. K., Gomez, L. M., & Dumais, S. T.: The Vocabulary Problem in Human-System Communication. Communications of the ACM. 30 11 (1987) 964-971
11. Greenberg, S. & Rouding, M.: The Notification Collage: Posting Information to Public and Personal Displays. In: Proceedings of CHI 2001 ACM Press (2001) 514-521
12. Gregory, R. L.: Eye and Brain. Princeton University Press, Princeton, NJ (1997)
13. Grudin, J.: The Case Against User Interface Consistency. CACM. 32 10 (1989) 1164-1173
14. Grudin, J.: Partitioning digital worlds: focal and peripheral awareness in multiple monitor use. In: Proceedings of CHI 2001 (2001) 458-465
15. Hall, W.: Mostly Linkless. In: Proceedings of Hypertext'01 ACM Press (2001) 3
16. Harter, S. P.: Psychological Relevance and Information Science. Journal of the American Society for Information Science. 43 (1992) 602-615
17. Helms Jørgensen, A.: Thinking-Aloud in User Interface Design: a Method Promoting Cognitive Ergonomics. Ergonomics. 33 4 (1990) 501-507
18. Hertzum, M. & Jacobsen, N. E.: The Evaluator Effect: A Chilling Fact About Usability Evaluation Methods. Int. Journal of Human-Computer Interaction. 13 (2001) 421-443
19. James, W.: Principles of Psychology. Henry Holt & Co. (1890)
20. Jansen, B. J., Spink, A., Bateman, J., & Saracevic, T.: Real Life Information Retrieval: A Study of User Queries on the Web. SIGIR Forum. 32 1 (1998) 5-17
21. Lewis, C., Polson, P., Wharton, C., & Rieman, J.: Testing a Walkthrough Methodology for Theory-Based Design of Walk-Up-and-Use Interfaces. In: Proc. of CHI'90 (1990) 235-242
22. Molich, R.: Brugervenlige edb-systemer (in Danish). Teknisk Forlag (1994)
23. Naur P.: Human knowing, language, and discrete structures. (1988) In: Computing: A Human Activity, ACM Press/Addison Wesley (1992)
24. Naur, P.: Knowing and the Mystique of Logic and Rules. Kluwer Academic Publishers, Dordrecht, the Netherlands (1995)
25. Naur, P.: CHI and Human Thinking. In: Proceedings of NordiCHI 2000 (2000)

26. Naur, P.: Anti-philosophical Dictionary. naur.com Publishing (2001)
27. Naur, P.: Psykologi i videnskabelig rekonstruktion (Eng. Psychology in Scientific-Scholarly Reconstruction) (2001)
28. Nielsen, J.: Usability Engineering. Academic Press, San Diego CA (1993)
29. Nielsen, J. & Mack, R. L.: Usability Inspection Methods. Wiley and Sons Inc. (1994)
30. Nielsen, J. & Molich, R.: Heuristic evaluation of user interfaces. In: Proc. of CHI'90 (1990) 249-256
31. Norman D.: Some Observations on Mental Models. In: Gentner, D. & Stevens, A.L. (Eds.), Mental Models, Erlbaum, Hillsdale NJ (1983), 7-14.
32. Pinelle, D. & Gutwin, C.: Groupware walkthrough: adding context to groupware usability evaluation . In: Proceedings of CHI 2002 (2002) 455-462.
33. Pirolli, P. & Card, S.: Information Foraging. Psychological Review. 106 4 (1999) 643-675.
34. Pirolli, P., Card, S., & Van Der Wege, M.: The Effect of Information Scent on Searching Information Visualizations of Large Tree Structures. In: Proc. of AVI'2000 (2000) 161-172
35. Preece, J., Rogers, Y., & Sharp, H.: Interaction Design. John Wiley & Sons (2002)
36. Raskin, J.: The Humane Interface: New Directions for Designing Interactive Systems. Addison-Wesley, Reading MA (2000)
37. Sears, A. & Shneiderman, B.: Split Menus: Effectively Using Selection Frequency to Organize Menus. ACM Transactions on Computer-Human Interaction. 1 1 (1994) 27-51
38. Shneiderman, B.: Designing the User Interface. Addison-Wesley, Reading, MA (1998)
39. Smith, S. L. & Mosier, J. N.: Guidelines for designing user interfaces for software. ESD-TR-86-278 (1986) The MITRE Corporation
40. Somberg, B. L.: A Comparison of Rule-Based and Positionally Constant Arrangements of Computer Menu Items. In: Proceedings of CHI+GI'87 (1987) 255-260
41. Spencer, R.: The Streamlined Cognitive Walkthrough Method, Working Around Social Constraints Encountered in a Software Development Company. In: Proceedings of CHI'2000 (2000) 353-359
42. Wharton C. et al.: The cognitive walkthrough method: a practitioner's guide. In: Nielsen, J. & Mack, R. Usability Inspection Methods, John Wiley & Sons, New York NY (1994), 105-140