The Think Aloud Method and User Interface Design

M.W.M. Jaspers
University of Amsterdam, The Netherlands

INTRODUCTION

Daily use of computer systems often has been hampered by poorly designed user interfaces. Since the functionality of a computer system is made available through its user interface, its design has a huge influence on the usability of these systems (Carroll, 2002; Preece, 2002). From the user's perspective, the user interface is the only visible and, hence, most important part of the computer system; thus, it receives high priority in designing computer systems.

A plea for human-oriented design in which the potentials of computer systems are tuned to the intended user in the context of their utilization has been made (Rossen & Carroll, 2002).

An analysis of the strategies that humans use in performing tasks that are to be computer-supported is a key issue in human-oriented design of user interfaces. Good interface design thus requires a deep understanding of how humans perform a task that finally will be computer-supported. These insights then may be used to design a user interface that directly refers to their information processing activities. A variety of methodologies and techniques can be applied to analyze end users' information processing activities in the context of a specific task environment among user-centered design methodologies. More specifically, cognitive engineering techniques are promoted to improve computer systems' usability (Gerhardt-Powels, 1996; Stary & Peschl, 1998).

Cognitive engineering as a field aims at understanding the fundamental principles behind human activities that are relevant in the context of designing a system that supports these activities (Stary & Peschl, 1998). The ultimate goal is to develop end versions of computer systems that support users of these systems to the maximum in performing tasks in such a way that the intended tasks can be accomplished with minimal cognitive effort. Empirical research has indeed shown that cognitively engineered interfaces are considered superior by users in terms of supporting task performance, workload, and satisfaction, compared to non-cognitively engineered interfaces (Gerhardt-Powels, 1996). Methods such as the think aloud method, verbal protocol analysis, or cognitive task analysis are used to analyze in detail the way in which humans perform tasks, mostly in interaction with a prototype computer system.

BACKGROUND

In this section, we describe how the think aloud method can be used to analyze a user's task behavior in daily life situations or in interaction with a computer system and how these insights may be used to improve the design of computer systems. Thereafter, we will go into the pros and cons of the think aloud method.

The Think Aloud Method

Thinking aloud is a method that requires subjects to talk aloud while solving a problem or performing a task (Ericsson & Simon, 1993). This method traditionally had applications in psychological and educational research on cognitive processes. It is based on the idea that one can observe human thought processes that take place in consciousness. Thinking aloud, therefore, may be used to know more about these cognitive processes and to build computer systems on the basis of these insights. Overall, the method consists of (1) collecting think aloud reports in a systematic way and (2) analyzing these reports to gain a deeper understanding of the cognitive processes that take place in tackling a problem. These reports are collected by instructing subjects to

solve a problem while thinking aloud; that is, stating directly what they think. The data so gathered are very direct; there is no delay. These verbal utterances are transcribed, resulting in verbal protocols, which require substantial analysis and interpretation to gain deep insight into the way subjects perform tasks (Deffner, 1990).

The Use of the Think Aloud Method in Computer System Design

In designing computer systems, the think aloud method can be used in two ways: (1) to analyze users' task behaviors in (simulated) working practices, after which a computer system is actually built that will support the user in executing similar tasks in future; or (2) to reveal usability problems that a user encounters in interaction with a (prototype) computer system that already supports the user in performing certain tasks.

In both situations, the identification and selection of a representative sample of (potential) end users is crucial. The subject sample should consist of persons who are representative of those end users who will actually use the system in the future. This requires a clearly defined user profile, which describes the range of relevant skills of system users. Computer expertise, roles of subjects in the workplace, and a person's expertise in the domain of work that the computer system will support are useful dimensions in this respect (Kushnirek & Patel, 2004). A questionnaire may be given either before or after the session to obtain this information. As the think aloud method provides a rich source of data, a small sample of subjects (eight to 10) suffices to gain a thorough understanding of task behavior (Ericsson & Simon, 1993) or to identify the main usability problems with a computer system (Boren & Ramey, 2000). A representative sample of the tasks to be used in the think aloud study is likewise essential. Tasks should be selected that end users are expected to perform while using the (future) computer system. This requirement asks for a careful design of tasks to be used in the study to assure that tasks are realistic and representative of daily life situations. It is recommended that task cases be developed from real-life task examples (Kushnirek & Patel, 2004).

Instructions to the subjects about the task at hand should be given routinely. The instruction on thinking aloud is straightforward. The essence is that the subject performs the task at hand, possibly supported by a computer, and says out loud what comes to mind.

A typical instruction would be, "I will give you a task. Please keep talking out loud while performing the task." Although most people do not have much difficulty rendering their thoughts, they should be given an opportunity to practice talking aloud while performing an example task. Example tasks should not be too different from the target task. As soon as the subject is working on the task, the role of the instructor is a restrained one. Interference should occur only when the subject stops talking. Then, the instructor should prompt the subject by the following instruction: "Keep on talking" (Ericsson & Simon, 1993).

Full audiotaping and/or videorecording of the subject's concurrent utterances during task performance and, if relevant, videorecording of the computer screens are required to capture all the verbal data and user/computer interactions in detail. After the session has been recorded, it has to be transcribed. Typing out complete verbal protocols is inevitable to be able to analyze the data in detail (Dix et al., 1998). Videorecordings may be viewed informally, or they may be analyzed formally to understand fully the way the subject performed the task or to detect the type and number of user-computer interaction problems.

The use of computer-supported tools that are able to link the verbal transcriptions to the corresponding video sequences may be considered to facilitate the analysis of the video data (Preece, 2002).

Prior to analyzing the audio and/or video data, it is usually necessary to develop a coding scheme to identify step-by-step how the subject tackled the task and/or to identify specific user/computer interaction problems in detail. Coding schemes may be developed bottom-up or top-down. In a bottom-up procedure, one would use part of the protocols to generate codes by taking every new occurrence of a cognitive subprocess code. For example, one could assign the code *guessing* to the following verbal statements: "Could it be X?" or "Let's try X." The remaining protocols then would be analyzed by using