Univ Access Inf Soc (2003) 2: 265–279 / Digital Object Identifier (DOI) 10.1007/s10209-003-0056-8

Online help for the general public: specific design issues and recommendations

A. Capobianco, N. Carbonell

LORIA, UMR 7503 (CNRS, INRIA, Universités de Nancy), Campus Scientifique, BP 239, 54506, Vandœuvre-lès-Nancy Cedex, France; {Antonio.Capobianco,Noelle.Carbonell}@loria.fr

Published online: 8 October 2003 –

Abstract. This paper addresses the issue of how to design online help that will really prove effective, accessible, and usable for all categories of users in the coming Information Society and, most of all, that will actually be used by novice users. The paper demonstrates the intrinsic necessity of online help and the actual failure of approaches claiming that "transparent" user interfaces eliminate the need for online support chiefly on the grounds that they encourage exploration. Empirical results in the literature or stemming from analyses of data we collected are put forward in the discussion. Based on a brief survey of the relevant literature, the major specific design issues that designers of online help systems are confronted with are presented, existing design approaches that might contribute to solving these issues are discussed, and a realistic short-term approach for improving the accessibility, effectiveness, and usability of online help systems is recommended. Our recommendation is mainly based on the results of a recently performed experimental study. These results led us to advise, at least for the near future, the design of noncontextual help systems for improving the accessibility, effectiveness, and usability of online help, rather than the implementation of dynamic adaptation to the current user's cognitive profile or the development of contextual help systems that generate the information content of help messages dynamically according to the user's current intention and goal. We assume that it is possible, within the framework of universal design principles, to significantly enhance the effectiveness and usability of standard noncontextual help systems, mainly by making the most of the recent advances in research on multimodal interaction, especially on the integration of speech into input modalities.

Keywords: Online help – Contextual help – User support – Universal design – Universal access – Adaptive systems

1 General introduction

Online help for the general public is still highly unsatisfactory. Despite the continuous efforts of researchers and designers over the last 20 years, most novice users still prefer consulting experienced users to browsing available online help or paper user guides.

This major weakness of available online help systems is becoming critical, as the implementation of concepts such as "pervasive" or "ubiquitous" computing, "ambient intelligence," or "smart" artifacts is developing at a fast pace, and the prospect of an overall Information Society is becoming a reality.

This paper addresses the crucial issue of how to design online help that will actually be used by the general public or, in other words, how to increase the accessibility, effectiveness, and usability of online help.

So far, current research on accessibility and universal access has been focused mainly on providing users with special needs (e.g., users with disabilities) with easy computer access; see, for instance, the design guidelines and standards proposed for implementing this concept [21, 42, 50]. More recently, the growing diversity of contexts of use (e.g., mobile computing and embedded systems) has motivated an increasing number of studies focused on the design of user interfaces taking advantage of new interaction devices and environments.

We claim that addressing issues relating to the diversity of individual cognitive capabilities and strategies may also contribute to promoting and implementing universal access in the coming Information Society.

In the first section, we demonstrate the necessity of online help for all standard user categories in Shneiderman's coarse taxonomy based on computer knowledge [39]. This taxonomy comprises three classes only: novice, occasional, and expert computer users, the bulk of the general public belonging to the first two classes.

The second section is focused on reviewing the specific challenges and issues confronting designers of interactive online help.

In the last section, we present some experimental results and discuss their possible contribution to the design of "accessible" online help systems, i.e., systems that will be able to satisfy the needs and expectations of all user categories included in the general public despite the great diversity of individual cognitive characteristics and, foremost, systems that will actually be used.

The discussion draws upon findings reported in the literature, together with empirical data we collected within the framework of two recent studies.

2 Online help, an essential user support

Since the beginning of the 1980s, research on user guides (or manuals) and online help has been developing along two parallel directions. One approach, which motivated many studies in the early 1980s when personal computers were becoming popular, is to design intuitive "transparent" user interfaces, that is, interfaces aiming at making interaction with standard application software so easy as to eliminate the need for initial training for their use and operation.

The other main approach, which has been motivating continuous research since the mid-1980s, is to provide novice users with easier access to the body of available help knowledge and to improve the relevance of the information they receive in answer to their requests by resorting to various software techniques, namely:

- by organizing help information in the form of databases or hypertexts; and, more recently,
- by affording users more "natural" or "intelligent" means of interaction, such as natural language dialogue facilities, adaptable or adaptive user interfaces.

However, the behaviors of novice users, especially users in the general public, may explain the failures of both research directions.

The limitations of direct manipulation for achieving transparency and the intrinsic necessity of assistance in using new application software are demonstrated in the three following subsections, using both empirical evidence published in the 1980s and analyses we performed on a corpus of expert-novice dialogues [3].

2.1 Direct manipulation: the quest for elusive "transparency"

One of the main motivations at the origin of direct manipulation was that intuitive, "transparent" user interfaces would eliminate the need for specific initial instruction and training for the use of new application software [39]; see also [10, 26, 28]. The basic argument put forward in support of this assumption can be summarized as follows.

Direct manipulation was created with a view to providing the "lay user" with interfaces more suited to their actual needs than artificial command languages [38]. Command languages, which were then the prevailing interaction paradigm, are characterized by complex syntaxes and arbitrary vocabularies.

Advocates of direct manipulation, such as Shneiderman [39], rightly assumed that novice and occasional users in the general public will have difficulty in learning and retaining how to use artificial command languages, especially in the absence of regular practice; these users will also be reluctant to acquire the necessary syntactical knowledge and therefore refrain from using such languages. On the other hand, the learning effort required to master their syntax is not likely to deter expert computer users from using them, the acquisition of syntactic knowledge being easier for frequent users. In addition, efficiency being one of the major concerns of expert users, the conciseness and expressive flexibility of command languages are likely to appeal to them.

Another assumption underlying the design of direct manipulation [38] was that intuitive manipulation of meaningful graphical representations of application objects, thanks to the implementation of an appropriate metaphor, was an attractive alternative to command languages, inasmuch as it would eliminate the necessity of an initial learning stage dedicated to the acquisition of appropriate syntactical knowledge. Users in the general public would be able to infer how to manipulate these representations (or icons) from their a priori knowledge of the application domain. Therefore, they would be able to discover and master the operation of any new application software by themselves simply through interacting with its user interface and exploring its functionalities; they would not need any explicit preliminary instruction.

The latter assumption is still common among today's interface designers and may contribute to explain why, at least for the time being, the design of new human-computer interaction paradigms raises more interest in the research community than the design of appropriate online help.

However, approaches aiming at achieving "transparent" human-computer interaction present limitations that are discussed in the next subsection. Both empirical results (see Sect. 2.2.1 and Sect. 2.2.2) and theoretical considerations (see Sect. 2.2.3) are put forward in support of our point of view.

2.2 Evidence pointing to the necessity of online help

2.2.1 Early empirical studies

Already in the 1980s a few empirical studies showed the illusory character of the assumption that "transparent" interaction could be achieved thanks to direct manipulation.

These studies demonstrated that, contrary to expectations, users did not actually take advantage of the exploration facilities offered by direct manipulation [9, 19]. Tentative interpretations are proposed for explaining these behaviors and attitudes in Sect. 3.

In addition, Streitz [44] pointed out that the office metaphor underlying direct manipulation could induce users to develop erroneous representations of the functionalities of a new software. For a discussion of the limitations of visual metaphors and direct manipulation, see [1].

Other studies demonstrated that users in the general public are unable to gain by themselves a sufficient knowledge of the operation of a new software for achieving the tasks they have in mind.

For instance, according to Mack et al. [26], novice users are confronted with numerous cognitive difficulties that they cannot overcome alone. In particular, they are unable to elaborate by themselves a reliable representation, or "system image" [45], of the capabilities and functioning of a new software. Faulty or incomplete representations are the source of many errors (semantic errors) that novices can neither detect nor correct for lack of the appropriate knowledge.

However, automatic prevention and correction of semantic errors is most difficult, cognitive interindividual diversity being too high to be fully taken into account by designers. Compatibility between the implementation of the designer's mental representation of the task domain (in the form of software functions) and the mental representation of this domain developed by each member of a large heterogeneous community of users such as the general public seems difficult, or even impossible, to achieve, a priori knowledge, cognitive capabilities, skills, and experience being indeed bound to vary greatly from one user to another in such a user community. In addition, designers being experts, their representation of the task domain and the system is bound to be drastically different from that of novice and occasional users (according to Shneiderman's taxonomy of users based on computer knowledge). This representational discrepancy, together with the great diversity of beginners' cognitive capabilities and experience, makes it extremely difficult to apply concepts such as adaptability (i.e., customization) or adaptivity to the design of user interfaces appropriate for novices; see Sect. 4.1 for a precise definition of these terms and a discussion of the usefulness of the concepts and techniques they refer to for the design of accessible and effective online help.

In practice, then, beginners have to "learn" and assimilate the conceptual representation of the task domain implemented by the designer (i.e., the semantics of the available functions) and adapt their behaviors to this representation. It follows that one of the main roles of online help should be to facilitate such assimilation and adaptation, for instance by outlining to users the relationships between their representations of the tasks they intend the system to carry out and the functions or procedures meant to execute them.

However, online help designers are also confronted with the great diversity of users' cognitive capabilities, a priori knowledge, skills, previous experiences, and current intentions, all of which influence beginners' representations of the task domain and of the corresponding software functions. Therefore, establishing appropriate links between the actual semantics of the functions of a software and the current user's "system image" for every user in the general public represents a difficult design challenge.

2.2.2 Additional evidence: our first study

We collected empirical data that confirm Mack's findings, inasmuch as they suggest that the major difficulty encountered by novice users is to establish the necessary relations between their intentions or goals and the software functions or procedures they have to run for achieving them.

Analyses were performed on ten expert-novice dialogues where two experts were to help ten novices carry out 20 predefined text-processing tasks using Microsoft (MS) Word. Expert and novice were in different rooms and could communicate freely using an intercom. In the first condition (condition A), the experts could view the novices' screen, whereas in the second one (condition B) they could not. All novices were unskilled users of Windows and had never used Word (they had only gone through half an hour preliminary training in mouse manipulation and the use of Word).

Dialogues were recorded and transcribed (by orthographic transcription). Transcripts were segmented into speech acts labeled using specific taxonomies derived from those available in the literature [34]. This study will be further referred to as Study 1.

Forty percent of the experts' speech acts (see Table 1) and 57% of the novices' requests (see Table 2) dealt with procedural information or, in other words, were focused on the clarification of the relationships between the novices' representations of the prescribed tasks and the software functions and procedures available to achieve them (see also [3]).

These results suggest that expert-novice dialogues were mainly focused on clarifying the links between the novices' mental representations of the task domain and those of the designer as reflected in the software functions. They also confirm indirectly the existence of a significant gap between "system images" developed by novices and the designer's representation of the software functionalities or those of expert users, thus pointing to the inadequacy of novice users' semantic representations of the functionalities of a new software.

In addition, our results show that the use of so-called intuitive interaction techniques such as direct manipulation is a source of difficulties and errors for beginners.

Table 1. (Study 1): Distribution of the experts' speech acts according to our specific taxonomy.
Condition A: the experts can view the subject's screen.
Condition B: the experts cannot view the subject's screen.
("E" means "Expert", and "N" "Novice").
Percentages have been computed over all the experts' speech acts

Speech act category	Percentage
Instruction (condition B) *E: To align the address against the right margin, you have to ...*	40%
Information (condition A) *E: "Sir" is not centered, it is inset 2cm from the left margin.*	27%
Assessment (condition A) *N: Do I use this button for scrolling?* N points at the lift. – *E: Yes.*	17%
Request for information (condition B) *E: Is the text cursor on the left of "Sir"?*	8%
Task planning (condition A) *E: Now, you can justify the text of the letter.*	8%

Table 2. (Study 1): Distribution of the novices' requests according to our specific taxonomy.
Condition A: the experts can view the subject's screen.
Condition B: the experts cannot view the subject's screen.
("N" means "Novice").
Percentages have been computed over all the novices' requests

Speech act category	Percentage
How? (condition A) *N: How can I put the cursor back at the beginning of the text?*	57%
Requests	(37%)
– For validation (condition A) *N: If I put the cursor here and type CR, is it OK?*	24%
– For assessment (condition A) *N: Is my cursor at the right position?*	12%
– For guidance in task planning (condition A) *N: Can I shift it (the address) now?*	1%
What? (condition A) *N: What is the use of this icon?*	3%
Why? (condition B) *N: Why has it (a word in the text) been erased?*	3%

Some subjects had difficulty in manipulating the mouse (especially "drag and drop" actions) and in using menus or dialogue boxes. Almost one third of the information exchanges between novice and expert concern such difficulties, which were the source of many syntactical and semantic errors. This observation suggests an additional weakness of direct manipulation as a "transparent" interaction style.

2.2.3 Gibson's theory of "affordance"

Some intrinsic limitations of the application of Gibson's theory of "affordance" to user interface design may also account for the observed limitations of direct manipulation and be put forward in support of our claim that "transparent" interaction is a difficult, even impossible, aim to achieve.

Among the numerous definitions proposed for the concept of affordance in the literature, we have chosen Norman's short definition since it is easily applicable to interactive software design:

> *... the term affordance refers to the perceived and actual properties of the thing, primarily those fundamental properties that determine just how the thing could possibly be used.* [31]

Application software can accurately reproduce only a limited range of human activities due to significant differences between human and computer communication or processing capabilities as well as between the properties of the real physical world where human activities take place and the characteristics of present computer-based environments. These differences represent major obstacles to the implementation of affordances. To overcome these obstacles, designers of current graphical interfaces resort to metaphors (e.g., the "office" metaphor underlying direct manipulation) and natural language (e.g., menu items and options in dialogue boxes). However, the semantic scope of any metaphor is intrinsically limited (similarity differing from identity). In addition, natural language is inherently ambiguous (due mainly to lexical polysemy and syntactic ambiguity). Finally, the great diversity of potential users' skills, knowledge, and experience is a hindrance to the success of this approach.

To sum up the discussion, let us quote Weiser and Seely's appraisal of the potential contribution of affordances to the design of "transparent" or "natural" human-computer interaction, namely:

> *An affordance is a relationship between an object in the world and the intentions, perceptions, and capabilities of a person. ... The idea of affordance, powerful as it is, tends to describe the surface of a design. For us the term 'affordance' does not reach far enough into the periphery where a design must be attuned to but not attended to.'* [49] http://www.ubiq.com/hypertext/weiser/calmtech/calmtech.htm

2.2.4 Summary of empirical findings

The empirical findings presented in this section, together with the limitations inherent in approaches aiming at achieving "transparent" human-computer interaction, lead to the conclusion that the design of intuitive human-computer interaction is a difficult aim to achieve. One can wonder whether it is even possible to design user interfaces that require no learning or adaptation effort from the novice user.

Without effective online help, novice and occasional users in the general public may reject a new software or artifact after a few trials, or even shy away from using it. Moore and Swartout observed that many novices experience stress during their first interactions with a new software [29]. Because stress is a major source of errors, it may be responsible for the numerous early rejections of new software applications intended for the general public that have been observed and reported in the literature.

Even knowledgeable users need online support. Online help may spare them the exploration (by trial and error) of the semantics of a new function and assist them in optimizing their interaction with the software, in addition to reminding them of syntactic details.

Online help appears, then, to be a necessity for all users, whatever their cognitive capabilities, knowledge, skills, and goals. To be accessible to all potential users, future "smart" artifacts and services will have to include online help facilities.

It follows that online help has the potential to improve software accessibility significantly, thus contributing to the implementation of universal access, provided that it proves flexible enough for helping all novice users, despite the diversity of their cognitive profiles, to efficiently overcome the difficulties they are confronted with while discovering a new software or function.

However, this potential has yet to be implemented. The major specific difficulties that hinder the implementation of effective online help are presented in the next section, and design approaches that may overcome them are tentatively proposed and discussed in the third and final section.

3 Online-help-specific design issues and challenges

Compared to the design of standard application software, user-centered design of online help must overcome additional specific obstacles originating mainly from the specific features of:

- the task involved, that is delivering to users the information they actually need just when they need it;
- the interaction situation, in other words, the necessary simultaneous involvement of users in two different, although related, activities, task execution, and help consultation;
- novice users' behaviors and attitudes.

These specificities are detailed in the three following subsections.

3.1 Accessibility and intelligibility of online help messages

Providing novice users of an application software with the metainformation they need to carry out the tasks motivating their use of the software is a specific challenge that help system designers must overcome.

Metainformation and metacommunication refer here to all exchanges of information relating to the conceptual model of the application software functionalities. This concept and its implications for human-computer interaction were proposed and discussed in the 1980s [46, 48], although, unfortunately, they exerted no sensible influence on user interface design.

As novice users are unable to elaborate by themselves an accurate operational representation of the software functionalities (see Sect. 2.2.1), they have difficulty in:

- "finding" the appropriate specific metainformation they need to complete the current interactive task and
- "understanding" generic help messages elaborated in terms of the conceptual model of the software functionalities.

In addition, they may have difficulty in efficiently using the help system itself, which represents for them another new software to be used simultaneously with the new application software. See [23, 37] for detailed analyses.

To overcome these difficulties, novice users of a given application software need specific online support. Such a support facility should be capable of providing them with easy access to the specific metainformation they need just when they need it. According to Waern [48], the main objective of this support should be to introduce the application software, to handle failures caused by the user, and to provide information on the current interaction context and the user's previous actions. Solutions to this major specific design issue will be proposed and discussed in Sect. 4.

Intelligible help messages are also essential for achieving easy successful metacommunication. For instance, numerous guidelines for the design of textual messages have been proposed; they advocate concise wording, limited message size, and selection of usual terms rather than technical jargon [11].

3.2 Interaction constraints

Another major specificity inherent in online help situations is that users must interrupt their main activity to interact with the help system. Switching back and forth between the application software and the related help system is an important source of difficulties, especially for novice users [23, 37].

Multimodality may be used to reduce the cost of these necessary switches. For instance, speech can be used to communicate with the help system and standard direct manipulation (or gestures combined with visual displays) to interact with the application software. Assigning different modalities (or sets of modalities) to each type of information (help vs. application) and each message direction (from user to application or help, or vice-versa) seems essential for obtaining a significant reduction in the user mental workload and for sensibly increasing the efficiency of the global interaction.

We performed a Wizard of Oz experiment (further referred to as Study 2) with a view to assessing the relative usability and effectiveness of two online help strategies, a contextual one and a noncontextual standard one. Subjects had to perform predefined text-processing tasks on a given text using MS Word. These tasks, which are presented in the appendix, were similar to the ones in our first study (see Sect. 2.2.2) in order to facilitate the implementation of the help strategies practiced by the human experts involved in this earlier study. Eighteen volunteer students performed these tasks in the course of a single session (lasting half an hour on average) during which they used successively each simulated help system. Subjects interacted with the help system using speech while help messages were displayed on an additional screen. Results of the analyses of postexperiment questionnaires and debriefing interviews indicate that the speech facility provided for accessing both help systems was a feature highly appreciated by a number of subjects. One third of them indeed expressed spontaneously their subjective satisfaction regarding this facility, either under the "Free comments" heading in the questionnaire or during their individual debriefing. This finding is all the more noteworthy as neither the questionnaire nor the debriefing guidelines included explicit references to the speech facility.

As multimodality appears to be a satisfactory approach for significantly reducing the intrusiveness of online help, no other solution will be proposed in Sect. 4.

3.3 Novice users' behaviors and attitudes

To be efficient, usable, and actually used, online help must overcome one major obstacle, that is, the "*motivational paradox*" [10], which greatly influences lay users' behaviors and attitudes toward online help.

Carroll and Rosson [10] conclude from the analysis of various empirical data that the general public, especially novice users, are reluctant both to explore a new software and to learn how to use its functionalities efficiently. These users concentrate from the start on achieving the tasks that motivate their interaction with a new software; mastering its use does not seem to rank among their (major) motivations or goals. Therefore, they are not likely to take advantage of the exploration facilities offered by direct manipulation. More generally, they are likely to ignore the facilities for autonomous learning, especially learning by trial and error, provided by intuitive user interfaces.

This motivational paradox, which, according to Carroll and Rosson [10], explains the reluctance of novice users to consult online help, is the most crucial specific difficulty that online help designers must overcome.

This paradox may also be held responsible for the failure of approaches that view online help as an interactive learning situation (e.g., [27]). In particular, it may account for the tendency of the general public to ignore online tutorials [8].

Finally, it may account for their reluctance to consult online manuals, even in the form of:

- databases [19, 36], although associative access to help information by means of simple keywords is easier and more intuitive than hierarchical access, the only available method for consulting standard manuals; or
- hypertexts [12, 29] and hypermedia documents [20, 33], even though these presentation techniques implement a more sophisticated associative access method than standard databases; links may indeed be viewed as keywords embedded in textual, or multimedia, contextual information, hence as contextual keywords.

A possible solution to this specific design issue, according to the authors of the minimum manual [11], is to give novices the means to achieve the tasks they have in mind as soon as they begin using the new software. The implementation of this strategy implies providing novice users with assistance at the task execution level or, in other words, developing human-computer cooperation. Therefore, this approach furthers the development of a "learning by doing" strategy on the part of novice users [8, 11].

3.4 Summary of design issues for help systems

To conclude, designers of online help for the general public are confronted with a number of specific issues that have been identified thanks to empirical studies dating back to the 1980s. These design issues stem mainly from:

- novice users' insufficient or erroneous knowledge of the conceptual model of the software functionalities (discussed in Sect. 3.1),
- the inherent intrusiveness of online help (see Sect. 3.2), and
- first and foremost, users' motivations (namely, the "motivational paradox" presented in Sect. 3.3).

Although these specific design difficulties have been addressed, they remain open issues. Today, online help systems for the general public are still ignored by users, whatever form they take: menu systems, databases, hypertexts or hypermedias, contextual messages activated by a mouse button (e.g., MS Excel), and anthropomorphic assistants (e.g., MS Word), among others.

To be able to overcome these difficulties, design approaches must consider online help as a specific human-computer interaction situation [15, 37].

In the next section, we present approaches and research directions that may improve the effectiveness of online help intended for the general public. We consider that, to be effective, online help must first and foremost actually be used, so that effectiveness, when applied to online help, subsumes the concepts of utility and usability as defined by Nielsen [30].

4 Toward the design of effective online help

In this section, we discuss the potential contribution of recent design approaches to overcoming the major specific difficulties and obstacles that confront online help designers as outlined in the preceding section.

The specific context of online help usage, that is, carrying out two simultaneous, distinct activities, namely, interaction with the software and retrieval of appropriate help information, will not be addressed here, as an exploratory solution yet to be experimented with and assessed has been proposed and discussed in Sect. 3.2.

From the discussion in the preceding section, it emerges that the main specific objectives that online help designers should attempt to achieve are:

- facilitate the acquisition of the knowledge and skills required to master the use of the application software without interfering with the novice's task-oriented current activity that motivates their interactions with the new software;
- enable all novices to gain such knowledge and skills easily, rapidly, and durably, using the potential of interactivity extensively (cf. [8]); this implies mainly:
  - taking into account the great interindividual diversity of their initial cognitive profiles (including cognitive capabilities, previous skills, experience, and knowledge); and,
  - helping them to develop an accurate representation of the functionalities of the new software (or "system image", see Sect. 2.2.1) while interacting with it;
- encourage a "learning by doing" approach by assisting them in carrying out the tasks they have in mind so as to overcome the "motivational paradox" (see Sect. 3.3).

The discussion below focuses on the application to online help design of the following concepts or paradigms: adaptability and adaptivity, universal design, and contextual help. Experimental results (stemming from Study 2) are presented to enrich the discussion of the advantages and limitations of contextual help.

The prevention and correction of errors will not be explicitly addressed in this section due to its specificity and complexity. However, any improvement in the effectiveness and usability of online help indirectly contributes to error prevention inasmuch as it reduces the global frequency of error occurrences.

4.1 Static and dynamic adaptation

Two major classes of techniques may be considered for adapting system responses to the current user's individual capabilities, knowledge, and preferences:

- *Adaptability*, based on predefined user categories or stereotypic static user models. Assignment of a stereotype or a category to the current user is performed either by the system or by the user (customization); see, for instance, [45] for a definition of stereotypes and a taxonomy of user models.
- *Adaptivity*, based on dynamic user models. Such models are capable of adapting to the current user's cognitive evolution in the course of interaction according to cues inferred from the analysis of their past actions (e.g., [18, 24]).

Obviously, a limited set of stereotypes or user categories cannot adequately represent the extent of interindividual cognitive diversity. Only adaptive models have the potential to accurately take account of a wide range of individual cognitive user profiles.

In addition, help systems must adapt to the progress of the current user in their use of a new software so as to account for the intraindividual cognitive variability inherent in learning situations. Such adaptation implies the capability of dynamically eliciting, and adapting to, users' current knowledge and skills.

Adaptive user interfaces have the potential to model the unbounded diversity of the initial cognitive profiles of novices as well as the evolution of their skills and knowledge in the course of interaction.

However, adaptivity is a difficult concept to implement when designing interactive systems that, like online help software, must adapt their behaviors to the cognitive evolution of the current user rather than to their preferences. The main difficulty is to design reliable algorithms capable of eliciting and interpreting accurately the contextual information (i.e., dynamic knowledge) needed to provide novices with appropriate adaptive support. This difficulty may explain why, so far, adaptivity has been mostly used for adapting Web user interfaces to the current user's preferences and browsing style; see [25] for a description of one of the first adaptive Web browser prototypes.

Sophisticated artificial intelligence techniques (uncertain reasoning, truth maintenance, etc.) have been implemented for modeling, from the analysis of the interaction history, the progress of novices in learning how to use a new software efficiently. However, the resulting online help prototypes (see, for instance, [2, 32]) are too complex and unreliable for meeting standard utility and usability evaluation criteria. In particular, novices are unable to form a clear and accurate mental representation of the system operation while interacting with such help systems because of their complexity.

The present state of the art may explain why adaptivity is so drastically limited in user interfaces to MS Pack Office software packages. Dynamic adaptation is indeed restricted to the following single straightforward feature: to obtain the display of all items in a menu, users have to click twice on the menu icon; otherwise only the functions they most frequently used previously are displayed. Research is still needed for achieving reliable and usable adaptive online help systems.

4.2 Universal Design

Universal design (UD) as defined in [13] appears to be an attractive alternative design paradigm for ensuring the usability and effectiveness of online help.

According to its creators and upholders, UD should be viewed as a fruitful concept and a useful body of experiences for promoting universal access in the forthcoming world-wide Information Society, since it focuses on:

> *... the design of products and environments to be usable by all people, to the greatest extent possible, without the need for adaptation or specialized design.* [43, page 4]

UD is an approach evolved by building designers, and it is not surprising that this definition and the seven principles that have been established to guide its implementation minimize the role and scope of adaptation.

A priori, UD principles are applicable to the design of online help systems and might prove a useful approach due to the fact that, despite the high diversity of their initial cognitive profiles, all novices have one feature in common: their ignorance of the software operation. This observation may justify the application of UD principles to the design of online help systems. The main asset of this approach is its simplicity. Robust implementations can be achieved easily within this framework, which makes it possible to use generic static user models rather than models that adapt dynamically to the characteristics of the current user and to their evolutions.

However, the UD paradigm does not explicitly address the specific online help design issue stemming from the "motivational paradox," which, according to Carroll and his colleagues (see Sect. 3.3), characterizes novice users' behaviors and attitudes toward online help. It is nevertheless possible to solve this issue within the UD paradigm framework and to enable novice users to develop a "learning by doing" approach. In order to efficiently assist novice users in achieving their task-related goals and intentions, online help systems must be capable of:

- inferring from the interaction context the information the user needs in order to complete the task in progress or realize the current goal, and
- delivering this information at the right moment, that is, right when users are in need of it.

Throughout the remainder of the paper, we shall use the adjective "contextual" to characterize this form of online help and apply the qualifier "noncontextual" to all the other online help systems, especially standard commercialized ones.

One of the first implementations of contextual online help is described by Dzida et al. [17]. We performed an experimental study, referred to in this paper as Study 2 (see Sect. 3.2), in order to assess the effectiveness of contextual help strategies. Results of this evaluation are presented in the next subsection, where we discuss the potential benefits and limitations of this design approach.

4.3 Contextual online help

4.3.1 Specific design issues

Help systems have to be aware of the goals and intentions that novice users are trying to achieve while interacting with a new application software in order to be able to assist them in carrying out the interactions necessary for completing the tasks they have in mind, that is, in order to provide them with contextual online help (see [6, 35] for sophisticated implementations of this form of help).

Analyses of the empirical data collected during our first study (see Sect. 2.2.2) suggest that human experts apply such a strategy, since:

- About two thirds of the experts' speech acts represent:
  - attempts at helping novices to achieve their goals and tasks or to plan their activity, or
  - assessments of the effects of users' actions on the interface in relation to the progress of the current task execution.
- Less than one third of their speech acts are general, nonprocedural explanations concerning the software functioning or the semantics of its functions.
- In addition, almost one half (46%) of experts' speech acts include references to:
  - the subject's current action on, or past interactions with, the software,
  - the current execution state of the task in progress, and
  - the current display, as indicative of the software current state.

However, the implementation of the experts' help strategies is a difficult challenge to take up. In particular, taking into consideration the current state of the art, it is difficult to achieve accurate automatic identification of the intentions that motivate users' actions, goals, and tasks. It is even more difficult than dynamically modeling, from the analysis of the interaction history, the novice's progress in learning how to use a new software efficiently.

Approaches based on pattern recognition techniques or statistical learning are basically inadequate. Cooperation with the novice user in order to promote "learning by doing" implies the capability of identifying the user's current intention and the task in progress it contributes to carry out, as early as possible, that is, after the first atomic actions in the execution procedure of the task have been carried out. In particular, novice users' intentions are difficult to identify because of frequent discrepancies between their intentions and the actions they perform to fulfill them. Early identification of these data is also necessary for the prevention of errors, which requires the capability of anticipating the user's next action or sequence of actions.

However, early accurate identification is often impossible due to the intrinsic ambiguity of the basic software functions, especially in the case of toolkits. For instance, one cannot infer from the selection of a graphical object on the screen which software function the user wants to apply to this object.

A simple reliable strategy for solving such ambiguities consists in inducing novice users to express their intentions and goals before attempting to achieve them. Both human experts who participated in our first study (see Sect. 2.2.2) often initiated clarification subdialogues in order to elicit the subjects' current intentions and goals.

The implementation of this approach and the assessment of its ergonomic quality are research subjects worth investigating. The control of such dialogues is straightforward since they are initiated by the system and their scope is limited. On the other hand, the interpretation of user's utterances may prove difficult due to the possible discrepancies between the novice or lay users' representations of the task domain and the actual functionalities implemented in the application software. Therefore, the system will have to "translate" intentions and goals expressed by novice users in terms of the software functionalities; this translation is bound to be complex due to the high interindividual diversity and intraindividual evolution of novice users' representations of the task domain.

To conclude, the elicitation of novice users' current intentions and goals is not to be considered as an insuperable obstacle to the implementation of contextual online help. It remains to assess the effectiveness of this form of online help, especially in comparison to standard noncontextual help, before further investigating implementation issues. The next subsection reports the results of our experimental evaluation of the actual utility and usability of contextual help strategies. This study, like the first one, involved potential users.

4.3.2 Evaluation of the effectiveness of contextual help

Our second study yielded quantitative results that show no marked differences between contextual and noncontextual help strategies as regards their effectiveness. After a brief description of our implementation of each strategy, results are presented and discussed. For a synopsis of the overall experimental protocol and set-up, see Sect. 3.2.

4.3.2.1 Contextual and noncontextual help messages

Procedural information on the operation of MS Word was presented textually and illustrated with screen copies of the relevant menu items and tool bars in both contextual and noncontextual messages. However, implementations of the two help strategies differed with respect to the presentation of execution procedures and, to a lesser extent, the information content of help messages.

bullet Noncontextual strategy:
The description of the whole procedure was included in one single noncontextual message that was fully displayed in answer to subjects' oral requests.

bullet Contextual strategy:
The procedure was split up into successive simple actions (e.g., selection of an icon), and a contextual message was associated with the description of each action. In answer to any request for a description of the whole procedure, the wizard displayed the first message in the sequence and waited until subjects had carried out the action described in the message before displaying the description of the following action. Preconditions and postconditions of recommended actions were specified in terms of requirements on the application displays.

Subjects performed half of the 18 predefined text-processing tasks using each simulated help system in counterbalanced order. All of them were familiar with Windows and Excel since the curriculum for first-year students in biology includes an introduction to Excel. However, most of them had never used Word. Those who had already used it had only a superficial knowledge of its functionalities; in particular, they had never used any of the functions that had to be activated for carrying out the 18 prescribed text-processing tasks.

4.3.2.2 Results: presentation and discussion

Two subjects did not go through all the prescribed text-processing tasks; they gave up after repeated failures. We excluded them from our analyses, which were then performed on the interactions of 16, instead of 18, subjects.

Quantitative results were computed from the following data:

- the percentage of tasks carried out successfully by subjects;
- the percentage of tasks they performed optimally, that is, according to the most efficient procedure, in particular, without resorting to trial-and-error strategies.

As the order of conditions (contextual then noncontextual help, vs. the reverse order) had no statistically significant influence on subjects' performances, measurements were grouped for each task into three sets according to the type of assistance subjects had used for performing the task: contextual (C) or noncontextual (NC) help or, since help was provided only on demand, no assistance (NH). Each set includes about one third of all task executions (32%, 34%, and 34%, respectively).

Global performance results

Results are presented in Table 3 in the form of percentages computed against all values in each set (C, NC, NH). Values of interset statistical comparisons (t-test) are reported in the rightmost column.

In keeping with our expectations, performances were markedly lower for the tasks performed without help than for those that subjects executed with the assistance of any of the two simulated help systems. However, surprisingly enough, there is no marked difference between the effectiveness of contextual and noncontextual help as regards successful task execution (barely significant difference) and use of the optimal realization strategy (no significant difference).

The influence of interindividual cognitive differences appears to be the only possible factor that may be put forward in order to explain these results convincingly and interpret them reliably.

Additional results – Cognitive tests

In order to validate this assumption, we analyzed the results of two cognitive tests that participants underwent before the experiment. These psychological tests were intended to assess the diversity of the subjects' cognitive capabilities and behaviors. We selected

- the ***GEFT*** (Group Embedded Figures Test), which is meant to evaluate the ability to perceive an object in a visual scene and isolate it from its context; according to Huteau [22], GEFT results are correlated with the capacity to solve problems using an analytical strategy. In the context of human-computer interaction, this test has been used to characterize interaction styles [14, 16]: higher scores characterize users who favor exploration, whereas lower scores denote reluctance to exploration together with an aptitude for processing linear directions efficiently; and
- the ***BLS4*** test, which provides a global assessment of the mental faculties that make up intelligence and, therefore, are involved in most cognitive activities [41]; for a description of the main constituents of intelligence, see [7], for instance. We used the BLS4 test to assess the subjects' ability to understand help messages, that is, to accurately interpret and carry out the instructions included in them.

Statistical results presented in Table 4 were obtained by grouping together, for each subject, the tasks they had performed in each condition (i.e., contextual help vs. noncontextual help). These results indicate that individual exploration capacities, as measured by the GEFT, significantly influenced successful task execution rates in either condition and, to a lesser extent (simple trends), optimal realization rates.

Table 3. (Study 2): Subjects' performances (task completion, realization strategy, execution time) according to the type of assistance provided (contextual, noncontextual, none). Significant results (t-test) are in bold type, trends in italics

Measurement	Contextual help	Noncontextual help	No help	Statistical significance
Success (%)	92	81	56	NH vs. NC: **p=0.00206** NH vs. C: **p=0.00025** NC vs. C: **p=0.04185**
Optimal strategy (%)	65	59	35	NH vs. NC: **p=0.002206** NH vs. C: **p=0.000347** NC vs. C: *p=0.217367*

Table 4. (Study 2): Comparisons between the subjects' performances in the two conditions, contextual vs. noncontextual help. t-test results. Significant results are in bold type, trends in italics

	Contextual help		Noncontextual help	
Task execution	Successful	Optimal	Successful	Optimal
GEFT test	**p=0.00581**	*p=0.06220*	**p=0.01986**	*p=0.08782*
BLS4 test	**p=9.82 E-05**	p=0.14742	**p=1.47 E-06**	*p=0.09502*

Subjects' aptitudes for using help messages effectively (as evaluated by the BLS4 test) also exerted a significant influence on successful task execution rates in both conditions, but their influence on optimal realization rates was rather limited in the noncontextual help condition (simple trend) and nonexistent in the contextual help condition (no significance).

We performed further statistical analyses on successful task completion rates in order to characterize, in each condition, the statistically significant effects of individual cognitive characteristics on performance.

Results of correlation tests, which are presented in Table 5, suggest that:

- subjects who got high GEFT marks successfully completed more tasks while using contextual help than subjects who got low marks; and this correlation is significant (see Table 4). On the other hand, there is no correlation, in the noncontextual help condition, between interindividual differences in exploration capabilities and successful task execution rates.
- as for interindividual differences in intelligence, as measured by the BLS4 test, we observed a significant correlation between BLS4 marks and performance rates (successful task execution) in both conditions, the correlation coefficient being sensibly higher in the noncontextual help condition.

The second result may be explained easily if our assumption that BLS4 marks are correlated with linguistic understanding capabilities is correct. Noncontextual messages were longer and more complex than contextual ones; hence they were more difficult to interpret, and the useful procedural information they contained was less easy to isolate, assimilate, and put into practice.

The rather small difference between the two correlation coefficients may be interpreted as an indication that the content, wording, and presentation of help messages need further improvement in order to facilitate their interpretation by novice users and reduce demands on their general intelligence capacity.

On the other hand, the first result is somewhat surprising. As contextual help, unlike noncontextual help, provides users with some guidance in task execution, it might be expected that subjects would not have to draw upon their exploration capabilities while using contextual help, so that no significant correlation between GEFT marks and successful task execution rates would be observed in this condition.

Further empirical and experimental studies are needed to confirm this unexpected result and to satisfactorily explain it. We observed that, in the contextual help condition, subjects did not always read all the successive messages describing a procedure; they often contented themselves with reading the first one in the sequence. This behavior may result from what might be considered a flaw in our experimental protocol: the fact that every help message in the sequence of messages describing a procedure was displayed automatically as soon as subjects had completed the corresponding step. Therefore, subjects read the first message in such a sequence, started to carry out the included instructions, and then got so engrossed in carrying on with the execution of the current procedure that they often ignored the next message. Subjects with lower GEFT scores are more likely to adopt such a behavior than subjects with higher scores, inasmuch as alternating between two different activities, namely, task execution and help consulting, implies drawing upon the cognitive capacities measured by the GEFT. Consequently, the high correlation observed between GEFT marks and successful task execution rates in the contextual help condition may be ascribed to the effects of the strategy we adopted for displaying help messages in the contextual help condition. This strategy (i.e., procedure decomposition into atomic actions) may

Table 5. (Study 2): Correlations between subjects' cognitive profiles and their performances in the two conditions, contextual vs. noncontextual help. Significant results are in bold type, trends in italics

	Contextual help		Noncontextual help	
Task execution	Successful	Optimal	Successful	Optimal
GEFT test	**C=0.545**	$C = 0.753$	**C=0.013**	$C = 0.248$
BLS4 test	**C=0.265**	C=0.109	**C=0.377**	$C = 0.404$

have induced subjects to adopt behaviors likely to affect both successful task execution and optimal realization rates.

Synthesis

On the whole, comparisons between the subjects' cognitive capabilities and their performances yielded useful results despite the weakness detected a posteriori in the experimental protocol. These results clearly indicate that contextual help is more effective than noncontextual help as regards successful task execution (statistical significance). As for optimal task realization, the superiority of contextual over noncontextual help is not demonstrated since the difference does not reach statistical significance (simple trend).

In addition, interindividual cognitive differences significantly influenced subjects' performances and, in particular, their successful task execution rates. Contrary to our expectations and, in all likelihood, due to a flaw in our experimental protocol, the cognitive capabilities measured by the GEFT influenced subjects' successful task execution rates in the contextual help condition, whereas they had no impact on subjects' performances in the noncontextual condition.

As expected, the influence of the capabilities measured by the BLS4 test on the subjects' performances were greater in the noncontextual help condition than in the contextual one, meaning that contextual help messages were easier to understand and put into practice than noncontextual ones. However, the difference between correlation coefficients was much smaller than the difference observed for the GEFT.

4.4 Conclusions

Despite the limitations of our second study, the main results obtained represent a valuable contribution to the general issue discussed in Sect. 4, that is:

Which of the three approaches presented in this section, namely, dynamic adaptation, universal design, and contextual user support, will prove most appropriate and efficient for designing accessible, effective, and actually used online help systems?

These experimental results lead us to favor and recommend the adoption of a universal design approach as defined in [13], hence to advise designers against approaches implementing any form of dynamic adaptation to users' profiles and/or activities, such as adaptivity or contextual help. This conclusion is based on the following arguments, which stem from the experimental data we collected and analyzed.

- The differences between the performances achieved by subjects in the two conditions, C and NC, are not very important, especially in comparison with their performances for the tasks they performed without any help (NH):
  - 11% (C-NC) for successful task execution vs. 25% (NC-NH)
  - and 6% (C-NC) for optimal task realization vs. 24% (NC-NH).

  In addition, whereas the comparisons involving the NH set are significant, those between C and NC barely reach statistical significance (successful task execution rates) or are not significant (optimal task realization rates).

  Therefore, considering the possible lack of robustness of contextual online help systems and the complexity of their design and development, it seems preferable, at least for the time being, to adopt a standard noncontextual design approach compatible with universal design principles. We then advise against resorting to design paradigms grounded on dynamic user and interaction models until robust methods and software tools are available for implementing adaptive or context-sensitive help strategies.
- Both simulated help systems in our study shared several features that are not yet implemented in standard commercialized help systems, namely:
  - display of help information on an auxiliary screen, thus reducing interferences between the subjects' main activity, text processing, and online help consultation (see Sect. 3.2);
  - speech-input facilities for requesting help information; this feature also contributed to reducing the interferences of help consultation with text processing; in addition, natural language provided subjects with a "transparent" interface to each help system;
  - finally, the information content of help messages and their wording; both were carefully thought out and illustrated with copies of menus and toolbars.

  These features may have greatly contributed to the effectiveness of both systems. In particular, they may explain why subjects spontaneously resorted to either help system while processing 66% of the prescribed tasks, even though most of them had never previously used any commercialized online help facility while interacting with application software intended for the general public.

We are currently studying the contribution of these features to the effectiveness of online help. In particular, we are considering the definition of an oral request language that would be both easy to use without specific training and easy to interpret reliably using present speech recognition systems or, in other words, that would avoid the pitfalls of both spontaneous speech and artificial computer languages. A method for defining such interaction languages as restricted subsets of natural language is described in [4] and further detailed in [5]; see also [40] for a different approach.

We are also revising the strategies we applied to design help messages with a view to improving their information content, wording, and presentation. Our main motivation stems from the fact that successful task execution rates are highly correlated with BLS4 results (statistically significant result) in the noncontextual condition, meaning that the interpretation and exploitation of noncontextual messages noticeably drew upon the subjects' general intelligence capabilities as measured by the BLS4 test.

To conclude, considering the present state of the art and the results of our second experimental study, we assume that noncontextual interactive help enabling novice or lay users to initiate simple information dialogues will be more effective than adaptive or contextual help systems, provided that the following constraints are satisfied:

- users should have the means to express oral requests quasi-spontaneously using an appropriate subset of natural language, and
- the system should be capable of generating multimodal help messages containing only the procedural information actually needed by users and presenting it nonintrusively in a form they can easily understand.

To achieve these goals, help systems must mainly be capable of bridging the gap between the layperson's standard representation of the task domain, which influences his intentions or goals, and the software designer's own representation, which determines the semantics of the software functionalities. Therefore, they must be familiar with both views and aware of the discrepancies between them. We assume that this knowledge does not need to be acquired dynamically by the system, at least for most applications intended for the general public, and that it consists in stereotypical a priori knowledge that can be implemented as static knowledge in the system. This last assumption implicitly hypothesizes that the diversity of novice users' representations of the task domain can be accurately modeled by a few stereotypes, despite the great variety of users' previous knowledge and experiences.

The experimental evaluation of the effectiveness of noncontextual help systems implementing the features mentioned above will enable one to determine whether this approach represents a satisfactory short-term alternative to adaptivity or contextual help for solving the major specific issues that designers of online help systems are confronted with (see Sect. 3).

5 Summary

This paper has addressed the crucial issue of how to design online help that will actually be used by the general public and will truly prove effective, accessible, and usable for all categories of users in the coming Information Society.

The first section of the paper has demonstrated the intrinsic necessity of online help and the actual failure of approaches claiming that "transparent" user interfaces eliminate the need for online support on the main ground that they encourage exploration. Empirical results in the literature or stemming from analyses of data we collected have been put forward in the discussion.

Based on a brief survey of the relevant literature, the second section has presented the major specific design issues that designers of online help systems are confronted with.

In the third section, we have discussed existing design approaches that might contribute to solving these issues and recommended a realistic short-term approach for improving the accessibility, effectiveness, and usability of online help systems. We advise, at least for the near future, the design of noncontextual help systems rather than the implementation of dynamic adaptation to the current user's cognitive profile or the development of contextual help systems that generate the information content of help messages dynamically according to the user's current intention and goal.

This conclusion stems from a survey of previous research and new experimental results, both indicating that:

- it is possible, within the framework of universal design principles, to significantly enhance the accessibility, effectiveness, and usability of standard noncontextual help systems mainly by making the most of the recent advances in research on multimodal interaction, especially on the integration of speech into input modalities; whereas
- significant scientific advances await fulfillment in order to be capable of designing and developing reliable dynamic help systems with reactions and behaviors that are easy to "understand" and to predict so as to avoid perplexing novice users in the general public.

However, this conclusion is tentative; further empirical or experimental research is needed to ascertain its validity.

References

1. Beaudoin-Lafon M (1997) Interaction instrumentale : de la manipulation directe à la réalité augmentée. Actes des neuvièmes journées sur l'interaction homme-machine. Cépaduès Editions, Toulouse, pp 97–104
2. Brajnik G, Tasso C (1994) A flexible tool for developing user modelling applications with nonmonotonic reasoning capabilities. Int J Hum Comput Stud 40(1):31–62
3. Capobianco A, Carbonell N (2001) Contextual online help: elicitation of human experts' strategies. In: Proceedings of HCI'01, New Orleans, August 2001, Lawrence Erlbaum, vol 3, pp 266–270
4. Carbonell N (2001) Recommendations for the design of usable multimodal command languages. In: Proceedings of HCI International 2001, New Orleans, 5–10 August 2001, Mahwah NJ, vol 3, Universal access in HCI: towards an information society for all. Lawrence Erlbaum, London, pp 266–270
5. Carbonell N (2003) Towards the design of usable multimodal interaction languages. Univ Access Inf Soc 2(2):142–158. (Special issue on multimodality: a step towards universal access)

6. Carenini G, Moore JD (1993) Generating explanations in context. In: Proceedings of IWIUI'93 International workshop on intelligent user interfaces, Orlando, January 1993, ACM Press, New York, pp 175–182
7. Carroll JM (1993) Human cognitive abilities: a survey of factor analytic studies. Cambridge University Press, Cambridge, UK
8. Carroll JM, Mack RL (1992) Learning to use a word processor: by doing, by thinking, and by knowing. In: Thomas JC, Schneider ML (eds) Proceedings of CHI'92, human factors in computer systems, ACM Press & Addison Wesley, New York, pp 13–51
9. Carroll JM, McKendree J (1987) Interface design issues for advice-giving expert systems. Communications of the ACM, 30:14–31
10. Carroll JM, Rosson MB (1987) Paradox of the active user. In: Carroll JM (ed) Interfacing thought: cognitive aspects of human-computer interaction. Bradford Books/MIT Press, Cambridge, MA, pp 81-111
11. Carroll JM, Smith-Kerker PL, Ford JR, Mazur-Rimetz SA (1987) The minimal manual. Hum Comput Interact 3(2):123–153
12. Cohill AM, Williges RC (1985) Retrieval of help information for novice users of interactive computer Systems. Hum Factors J Hum Factors Ergonom Soc 27(3):335–343
13. Connell BR, Jones M, Mace R, Mueller J, Mullick A, Ostroff E, Sanford J, Steinfeld E, Story M, Vanderheiden G (2001) What is Universal Design? Available at http://www.design.ncsu.edu:8120/cud/uni v_design/princ_overview.htm
14. Coventry L (1989) Some effects of cognitive style on learning UNIX. Int J Man Mach Stud 31(3):349–365
15. Duffy TM, Palmer JE, Mehlenbacher B (1992) Online help: design and evaluation. Ablex, Norwood, NJ
16. Dufresne A, Turcotte S (1997) Cognitive style and its implication for navigation strategies. In: Boulay BD, Mizoguchi R (eds) Proceedings of AI-ED'97, Kobe, Japan, August 1997, IOS Press, pp 287–293
17. Dzida W, Hoffmann C, Valder W (1987) Mastering the complexity of dialogue systems by the aid of work contexts. In: Bullinger HG, Kornwachs K, Shackel B (eds) Proceedings of the 2nd IFIP international conference on human-computer interaction, INTERACT'87, Stuttgart, September 1987, North Holland, Amsterdam, pp 29–33
18. Fink J, Kobsa A, Nill A (1997) Adaptable and adaptive information access for all users including the disabled and the elderly. In: Jameson A, Paris C, Tasso C (eds) Proceedings of the 6th international conference on user modeling, UM97, Chia Laguna, Italy, June 1997, Springer, Berlin Heidelberg New York, pp 171–173
19. Fisher G, Lemke A, Schwab T (1985) Knowledge-based help systems. In: Borman L, Curtis B (eds) Proceedings of CHI'85 (International conference on human factors in computing systems), San Francisco, April 1985, ACM Press & Addison Wesley, New York, pp 161–167
20. Harrison S (1995) A comparison of still, animated or nonillustrated on-line help with written or spoken instructions in a graphical user interface. In: Proceedings of CHI'95 (International conference on human factors in computing systems), Denver, May 1995, ACM Press & Addison Wesley, New York, pp 82–89
21. HEFS/ANSI (1997) Draft HFES/ANSI 200 Standard, Sect. 5: accessibility. Proceedings of Human Factors and Ergonomics Society, Santa Monica
22. Huteau M (1979) Style cognitif et pensée opératoire. Bull Psychol 33:12–15
23. Kearsley G (ed) (1988) Online help systems: design and implementation. Ablex, Norwood, NJ
24. Langley P (1999) User modelling in adaptive interfaces. In: Proceedings of the seventh international conference on user modeling, UM99, Banff, Canada, June 1999, Springer, Berlin Heidelberg New York, pp 357–370
25. Lieberman H, Maulsby D (1996) Instructible agents: software that just keeps getting better. IBM Sys J 35(3&4):539–556
26. Mack RL, Lewis C, Carroll JM (1983) Learning to use word processors: problems and prospects. ACM Transactions in office information systems, 1:254–271
27. Mallen LC (1995) Designing intelligent help within information processing systems. Ph.D. Thesis, Leeds University, UK
28. Mantei M, Haskell N (1983) Autobiography of a first-time discretionary microcomputer user. In: Proceedings of CHI'83 (International conference on human factors in computing systems), Boston, April 1983, ACM Press, New York, pp 286–290
29. Moore JD, Swartout WR (1990) Pointing: a way toward explanation dialogue. In: Proceedings of the eighth national conference on artificial intelligence, Boston, AAAI Press, Menlo Park, CA, 1:457–464
30. Nielsen J (ed) (1993) Usability engineering. Academic, San Diego
31. Norman DA (1988) The psychology of everyday things. Basic Books, New York
32. Paiva A, Self J (1994) A learner model reason maintenance system. In: Proceedings of ECAI'94, pp 193–196
33. Palmiter S, Elkerton J (1991) An evaluation of animated demonstrations for learning computer-based tasks. In: Proceedings of CHI'91 (International conference on human factors in computing systems), New Orleans, May 1991, ACM Press & Addison Wesley, New York, pp 257–263
34. Pilkington RM (1992) Question-answering for intelligent online help: the process of intelligent responding. Cognit Sci 16(4):455–491
35. Quast K-J (1993) Plan recognition for context sensitive help. In: Proceedings of IWIUI'93 (International workshop on intelligent user interfaces), Orlando, January 1993, ACM Press, New York, pp 89–96
36. Roestler AW, McLellan SG (1995) What help do users need? Taxonomies for on-line information needs and access methods. In: Proceedings of CHI'95 (International conference on human factors in computing systems), Denver, May 1995, ACM Press & Addison Wesley, New York, pp 437–441
37. Sellen A, Nicol A (1990) From users' task knowledge to high level interface specification. Int J Hum Comput Interact 6:1–15
38. Shneiderman B (1983) Direct manipulation: a step beyond programming languages. IEEE Comput 16:57–69
39. Shneiderman B (ed) (1987) Designing the user interface: Strategies for effective human computer interaction. Addison-Wesley, Reading, MA
40. Shriver S, Rosenfeld R, Zhu X, Toth A, Rudnicki A, Flueckiger M (2001) Universalizing speech: notes from the USI project. In: Dalsgaard P, Lindberg B, Benner H (eds) Proceedings of 7th European conference on speech communication and technology (EUROSPEECH'01), Aalborg, Denmark, 3–7 September 2001, ISCA, Bonn, Germany, 1:1563–1566
41. Spearman C (1927) The abilities of man. Macmillan, New York
42. Stephanidis C, Akoumianakis D, Ziegler J, Faehnrich K-P (1997) User interface accessibility: a retrospective of current standardisation efforts. In: Proceedings of HCI International'97. North Holland, Amsterdam, pp 469–472
43. Story MF (1988) Maximizing usability: the principles of universal design. Assist Technol 10(1):4–12
44. Streitz NA (1988) Mental models and metaphors: implications for the design of adaptive user-system interfaces. In: Mandl H, Lesgold A (eds) Learning issues for intelligent tutoring systems. Springer, Berlin Heidelberg New York
45. Sutcliffe AG, Old AC (1987) Do users know they have user models? Some experiences in the practice of user modelling. In: Proceedings of INTERACT'87. North Holland, Amsterdam, pp 36–41
46. Tauber MJ (1986) An approach to meta-communication in human-computer interaction. In: Klix F, Wandle H (eds) Proceedings of MACINTER1. North Holland, Amsterdam, pp 35–49
47. Waern Y (1985) Learning computerized tasks as related to prior task knowledge. Int J Man Mach Stud 22:441–455
48. Waern Y (1989) Cognitive aspects of computer supported tasks. Wiley, New York
49. Weiser M, Seely B (1995) Designing calm technology. Xerox Park technical report (21/12/1995) – Power-Grid J (Version 1.01, July 1996)
50. World Wide Web Consortium. Web Accessibility Initiative (1998) Available at http://www.w3.org/WAI/

Appendix

The 18 tasks participants in the second study had to carry out are summarized here. We only mention the MS Word functions or procedures that subjects had to execute to complete them; the actual scenarios are not reproduced.

1. Click on the ¶icon (that "displays" nonprintable characters)
2. Insert page break
3. Change text layout: from one column to several columns
4. Cut and paste text
5. Insert text – Insert blank line
6. Change style (using predefined styles such as "Title 1")
7. Justify text left
8. Change font face (e.g., from "Courier" to "Helvetica")
9. Change font type (e.g., from "Normal" to "Underlined")
10. Justify text left and right
11. Insert page numbers
12. Insert header – Center header – Change font size
13. Modify style definition
14. Search text – Replace text
15. Search and replace text
16. Change margin size
17. Frame text
18. Print document

In addition, at the beginning of the session, subjects were instructed to use the "Print Preview" function to visualize the results of their manipulations.