Language Learning & Technology
http://llt.msu.edu/vol2num1/article2/
July 1998, Volume 2, Number 1
pp. 40-53
(page numbers in PDF differ and should not be used for reference)

DESIGN AND EVALUATION OF THE USER INTERFACE OF FOREIGN LANGUAGE MULTIMEDIA SOFTWARE: A COGNITIVE APPROACH

Jan L. Plass
University of New Mexico

ABSTRACT

This paper is concerned with criteria for the design and evaluation of the user interface of foreign language multimedia software. A hybrid model is proposed that combines a cognitive and software engineering approaches. Based on this proposed contextualized model of interface design, domain-specific evaluation criteria are developed to describe how well the user interface is able to support the cognitive processes involved in the development of linguistic and pragmatic skills and competencies in SLA. The application of these criteria is demonstrated using the multimedia software *CyberBuch*/*Ciberteca*.

INTRODUCTION

The growing number of instructional multimedia software applications for SLA and the large variety of features and components of these programs generate a need for methods to evaluate systematically these materials. This paper is concerned with the design and evaluation of one of the most prominent components of a software product-- the user interface. Defined in very general terms as the part of an application in charge of communication with the learner, the user interface conveys the functionality of a computer application to the user, and translates the user's input into a machine-specific format (see Figure 1). Despite this key function of facilitating human-computer interaction, issues in the design of the user interface are often neglected in the development of instructional software. The approaches and criteria used by developers as a basis for interface design are often based more on intuition and experience than on theory-based models. While in many cases this may result in user interfaces of a high usability, it makes the development of systematic evaluation criteria for such systems difficult.

Attempts to define generally applicable design and evaluation criteria for multimedia software have resulted in a number of different approaches (Park & Hannafin, 1993; Ravden & Johnson, 1989). However, despite their comprehensive list of criteria these approaches are not specific enough to be usable for a particular subject matter area such as SLA. It is argued in this paper that evaluation criteria need to be developed based on domain specific learning processes and activities and on the cognitive processes that these activities involve. Using this approach, a taxonomy of SLA software features would be based on the underlying pedagogy or principles of adult education (andragogy) and activities and instructional methods of language learning and would address how well the individual components of the software are able to facilitate them.

In order to develop evaluation criteria for the user interface of foreign language multimedia software, I will first briefly review existing approaches to interface design and identify the specifics of multimedia applications for SLA. I will then propose a model for user interface design based on a cognitive approach and will apply this model to the CyberBuch/Ciberteca software (Chun & Plass, 1995, 1997b, 1998). From this proposed interface design model, I will derive evaluation criteria for the user interface of FL multimedia software with specific emphasis on reading instruction.

APPROACHES AND MODELS OF INTERFACE DESIGN

Using a definition of the user interface as the communication channel between the user and the functional elements of the computer (Furnes & Barfield, 1995; Marchionini, 1991; Waterworth, 1992), human-computer interaction can be seen as a system with three components: a computer/application, an interface, and a human user subsystems (see Figure 1).

The function of the interface subsystem is to assign user input to internal representations of the application and internal representations of the application to output that is comprehensible to the user. The type of input and output modes employed by the interface subsystem determines the type of the interface. For example a text-based system uses only the written verbal communication mode, whereas a direct manipulation system allows the user to manipulate objects and use visual, verbal, and auditory representations of the system state.

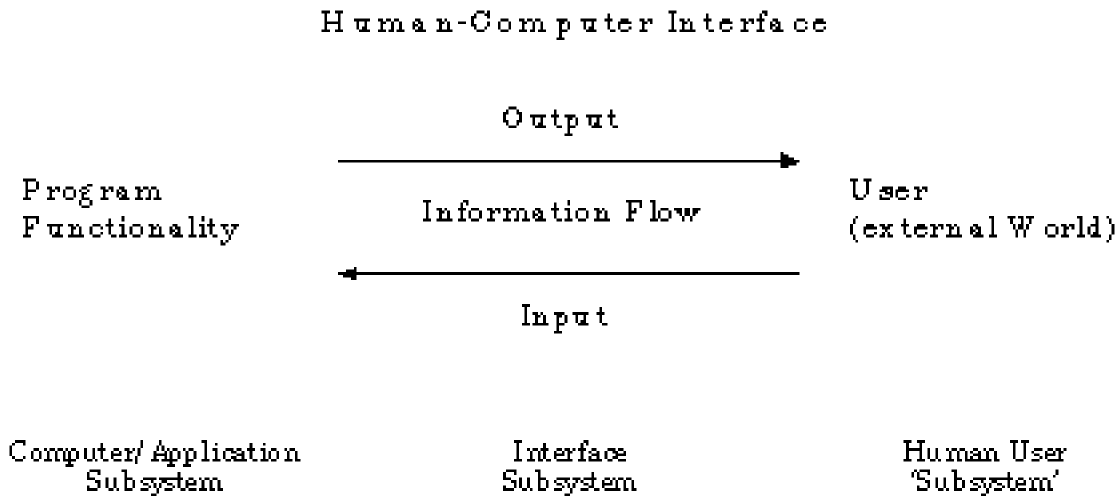

Figure 1. Definition of user interface

The design process of a user interface involves the development of a conceptual model of the application and its functionality by the designer, which is then implemented as the user interface, often using one or many metaphors. Users have to interpret the interface and build their own mental model of its functionality (see Figure 2).

The user interface developed by the designer of the application is influenced by the particular functionality of the software, which in turn is determined by the system's inherent structure. In some cases, this structure can influence the design of the user interface. For example, the text-based command-line interface of the DOS operating system contains design features that are influenced by the internal structure of the computer. The user has to learn a certain syntax of commands, parameters, and options that are closer to the machine code of a microprocessor than to natural language. This approach could be characterized as *Computer/Application-centered* design (Norman, 1990). The decision of how to implement the functionality conceptualized by the developer was based more on how the computer processes and stores information than on how the human user processes and stores it. This results in the requirement for the user to memorize procedural information that is irrelevant to the actual learning task but which is necessary for communication with the computer.

In contrast to *Computer/Application-centered* design, a *User-centered* design takes human factors into account. Here the human-computer interaction is designed with a focus on how humans process and store information. The goal is to allow the user to focus on the task at hand and reduce the amount of overhead knowledge required to communicate effectively with the computer. This approach requires extensive testing of the interface with actual users to study their behavior when using the software. Failure to conduct usability testing leads to the implementation of features that are solely based on the designer's preferences and intuition, which often results in inconsistent features that don't fit into the user's mental model of the application and its functionality. This approach could be characterized as *Designer-centered*. Therefore, the first question in developing evaluation criteria would be: Is the design user-centered? (Norman, 1990). In order to answer this question, we need to take a closer look at current design approaches for the development of the user interface.

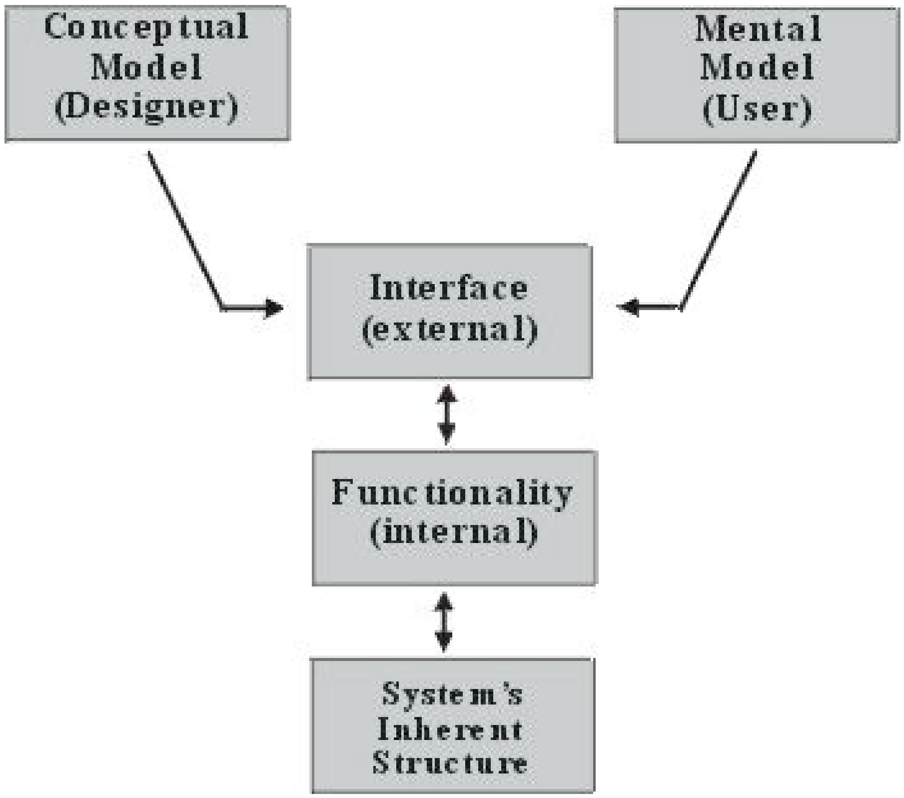

Figure 2. Design model for the user interface

In a more theory-based review of the current user interface design practice, Wallace and Anderson (1993) distinguish between four different types of approaches to user interface design:

- Craft approach
- Enhanced software engineering approach
- Technologist approach
- Cognitive approach

In the *craft approach*, the design is based on the skill and experience of the interface designer or human factors expert to suit the particular circumstances (Dayton, 1991; Laurel, 1990; Norman, 1987; Rubinstein & Hersh, 1984; Wroblewski, 1991). The goal of the design is to find the most appropriate features, based mainly on practical and economical considerations and the subjective judgment of the instructional designer rather than on a global dominating theory. The advocates of this approach view interface design as a craft, and put little emphasis on general principles underlying successful design. They argue that projects are so unique that the development of a structured methodology for interface design is impossible.

The *enhanced software engineering approach* incorporates human factors, such as user characteristics and task analysis, into traditional structured software engineering models exemplified by the waterfall model or the Jackson model (Damodaran, Ip, & Beck, 1988; Shneiderman, 1993; Sutcliffe, 1988, 1989; Waterworth, 1992; Winograd, 1992). The main focus of this pragmatic approach is on usability and the desire to serve the user effectively (Shneiderman, 1993).

The *technologist approach* focuses on providing software tools for interface design, aimed at automating and quantifying the design process (Buxton & Lamb, 1983: Cockton, 1988; Wasserman, 1985). Advocates of this approach stress the importance of rapid prototyping to identify user requirements, but do not regard the human-computer interaction expert as an important member of the design team. The design process is based on user interface management systems and the idea that good interfaces can be extracted from the user (Wallace & Anderson, 1993).

The *cognitive approach* applies psychological knowledge, such as theories of information processing and problem solving to interface design (Barnard, 1991; Card, Moran, & Newell, 1983; Gardiner & Christie, 1991; Kieras & Polson, 1985; Landauer, 1991). This approach is characterized by an attempt to measure the user's performance time and memory load for a given task, to identify prerequisite and acquired knowledge for a task, and to describe the user's mental models and mental processes for performing a task. The cognitive approach is the most theoretical approach to interface design, but it is often criticized for being too far removed from the practical needs of the interface designer.

Recognizing the weakness of an approach that is entirely context-free, a contextualized approach emerged that takes the specific content and procedures of a domain into consideration (Carroll, 1991; Dayton, 1991). Since the cognitive approach is the only one that puts both the user and the learning task in the center of the design process, it seems to be the most appropriate basis for the development of evaluation criteria

In summary, while there exist a number of different approaches and models of user interface design, only a few of them focus primarily on the learning process and the user. The existing approaches are either pragmatic and not firmly rooted in the theory of learning, or too complicated to be useful for practitioners of interface design. Moreover, no approach has been found that is specific to SLA and the instructional strategies and methods that are relevant to this field. In the next section, I will therefore summarize the specific considerations of foreign language software, and then integrate them into a cognitive approach to interface that is both theory-driven and pragmatic.

A COGNITIVE APPROACH TO INTERFACE DESIGN FOR FL MULTIMEDIA SOFTWARE

A *Cognitive Approach* appears to be the most appropriate basis both for the design and for the evaluation of user interfaces for SLA software since it incorporates both the user and the learning task into the design. In this section, I will describe some domain-specific issues of SLA, discuss the cognitive processes involved in SLA-related activities, and then propose a model for the design and evaluation of the user interface for these applications.

Specifics of FL Multimedia Software

Software for SLA can be designed in a variety of forms and styles and delivered in a variety of different ways, including CD-ROM-based or WWW-based instruction. What these different materials have in common is their goal of developing and improving linguistic and pragmatic skills and competencies (see Table 1).

Competencies/Skills	Examples of Learner Activities
Listening	Listen to passages (e.g., authentic conversations, news reports, literary texts)
Speaking	Record spontaneous speech; do intonation analysis and practice (Chun, 1998), use speech recognition tools (Ehsani & Knodt, 1998; Eskenazi, 1998; Price & Rypa, 1998)
Reading	Macro level: view visual advance organizers, read background information (Chun & Plass, 1996b); Micro level: look up multimedia annotations for vocabulary (Chun & Plass, 1996a)
Writing Composition	exercises, including peer-review, editing, and rewriting
Communicative	Real-time chat, e-mail exchange, discussion lists (Warschauer, 1997), use of speech recognition-based dialog systems (Luperfoy, 1998)
Sociolinguistic	Task-based, problem-solving, and role-playing activities that address sociolinguistic differences between native and target languages, and that could involve real-time chat, e-mail exchange, discussion lists (Chun, 1994)
Strategic	Task-based, problem-solving, and role-playing activities that require learners to achieve specific goals (e.g., persuading, self-correcting, negotiating a desired outcome); these could involve real-time chat, e-mail exchange, and discussion lists

Table 1. SLA Competencies / Skills and Learner Activities

To achieve these objectives, a variety of instructional activities are implemented that are supported by various tools and features of the program. The user interface of FL multimedia software has to facilitate the development of the particular linguistic and pragmatic skills and competencies that the software application addresses. The user interface design has to support the cognitive processes involved in these skills and competencies. Table 2 shows a list of cognitive processes generally involved in learning.

Table 2. Cognitive Processes Involved in Learning

Interpreting the performance goal
• Encoding or retrieving task-relevant declarative information • Compiling and executing procedural knowledge • Monitoring performance • Identifying sources of error in performance • Correcting errors in performance

The specific cognitive processes involved in SLA can be derived from the individual linguistic and pragmatic competencies listed above. In the case of reading skills, the process of text comprehension includes the following: (a) processing of prerequisite knowledge; (b) paying attention to and selecting relevant information; (c) building internal connections (i.e., reorganizing the new information in short term memory into a coherent form); (d) building external connections (i.e., integrating new information with the existing prerequisite knowledge into the learner's mental model) (Chun & Plass, 1997a; Mayer, 1997; Plass, Chun, Mayer, & Leutner, 1998; van Dijk & Kintsch, 1983). This is represented in Figure 3.

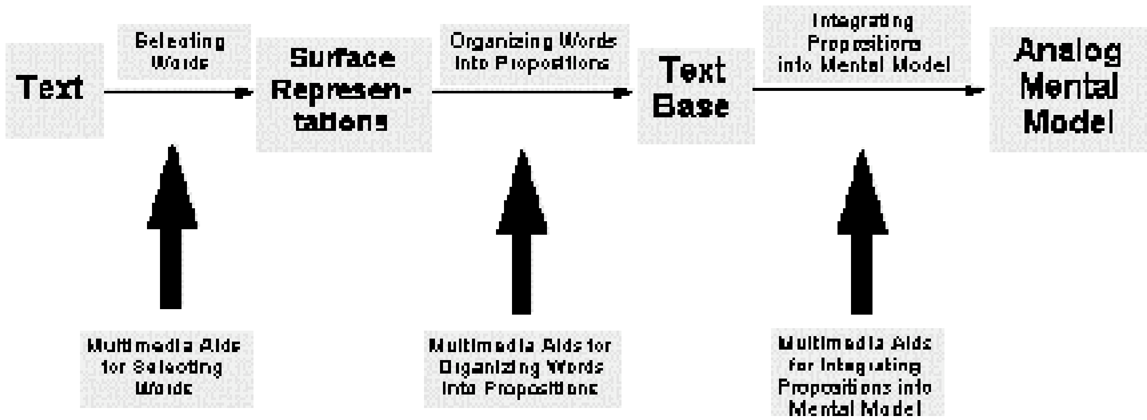

Figure 3. Cognitive processes in text comprehension and aids that support them

The function of multimedia tools is to aid the cognitive processes involved in the development of a particular linguistic and pragmatic competency (Chun & Plass, 1997a). Multimedia information can be used in a variety of functions (Levin, Anglin, & Cary, 1987).

- *Decoration* (motivational with little relation to instructional content)
- *Representation* (depiction of instructional content)
- *Organization* (depiction of knowledge structure)
- *Interpretation* (visualization of abstract concepts)
- *Transformation* (facilitation of higher-level cognitive processes)

These different functions of multimedia aids to text comprehension can be extended to all interface design elements. For example, properly designed navigational tools can aid in the cognitive processes of text comprehension by reducing the overhead information necessary to navigate the system. In other words, the design of instructional strategies that address specific competencies must be integrated with the design of the elements of the visual interface that support these strategies.

PROPOSED MODEL FOR INTERFACE DESIGN

The aforementioned function of the user interface leads to a new definition of the interface design process: Based on a cognitive approach, I propose the following definition:

Interface design is the process of selecting interface elements and features based on their ability to deliver support for the cognitive processes involved in the instructional activities facilitated by the application.

Based on this new definition, I propose a three-step model for interface design which is inspired by Clark and Sugrue's (in press) model of media selection and by the contextualized approach described in Wallace and Anderson (1993):

1) Select the instructional activity that supports cognitive processes of the competence or skill to be developed.
2) Select the attributes of the feature.
3) Select the design feature.

1. Select instructional activity that supports cognitive processes of competence or skill to be developed.

Based on a needs assessment, learner analysis, task and content analyses, and determination of goals and objectives, the design of instructional materials will include the selection of instructional methods with a number of activities to be performed by the learners. For second language instruction, the goal is the development of some or all of the linguistic and pragmatic competencies listed in Table 1, which is accomplished by selecting the appropriate instructional activities.

The type of activity selected will depend on the instructor's general instructional philosophy and on the specific circumstances and needs of the learners, but it is mainly determined by the objectives of the instruction and is, therefore, domain-dependent. For instance, science classes may include problem solving activities, whereas language classes might prefer communicative activities. These activities should support the cognitive processes involved in the specific competency or skill. For example, in the case of reading comprehension, the cognitive process of activating prior knowledge could be supported by the instructional activity of using an advance organizer. The process of building a text base from a text and organizing information in short-term memory could be supported by providing annotations for vocabulary items. It should be mentioned at this point that the selection of instructional methods and activities is also a basis for the selection of the delivery medium of the instruction (Clark & Sugrue, in press). This does not necessarily imply that the delivery medium has to be a software application.

2. Select attributes of feature.

After selecting the instructional activities to be implemented, the attributes of the interface features can be determined. Attributes in this context are properties of the design feature that have relevance for the effectiveness of the instruction. They include the functionality and visual appearance of both the feature and the application as a whole. These attributes can be derived from cognitive and educational psychology regarding human memory, attention, interest, motivation, processing of information, individual differences, and construction of mental models.

In the case of reading comprehension, the use of an advance organizer to support the process of activating prior knowledge would require attributes of this feature such as adaptability to different levels of prior knowledge, and ease of comprehension for learners with low prior knowledge. The use of annotations for vocabulary items to aid organizing information in short-term memory would require easy access to different types of annotations in different presentation modes, avoiding distraction from the reading process if the annotations are not needed, and avoiding covering the text when the annotation is displayed. Furthermore, the selection of the attributes of the design feature has to take the interaction of the different features of the application into consideration.

3. Select design feature.

Based on the selected instructional activity and the attributes of the design feature chosen, the interface designer now selects the actual feature and the form of its implementation. At this point, only the feature and its attributes have been selected and usually several different possibilities exist for the actual implementation. Interface designers and graphic designers can implement the feature based on their approach and on such constraints as cost. In the case of the above examples, the advance organizer was implemented as a preview movie with a voice-over that could be selected before the story was read. Four types of annotations were implemented: definitions in L1, translations in L2, and pictures and video visualizing the word. In addition, the pronunciation of the word was given using a sound file. The actual implementation is shown in Figure 4.

Figure 4. Example of annotations for vocabulary items in the project *Ciberteca*

The proposed approach to interface design puts the user, the content, and the instructional activity in the center of the design process. It incorporates theories from cognitive psychology but it is domain-specific enough to be practical. This model is not inherently prescriptive, and can be used to derive guidelines for a particular project. In this sense, it incorporates ideas from both the craft and the cognitive approaches. Furthermore, this approach is based on instructional systems design models (Anglin, 1995; Smith & Ragan, 1993) that correspond to the structured software engineering models from the enhanced software engineering approach. Finally, it allows for the use of CASE (computer-assisted software engineering) tools and the rapid prototyping method from the technologist approach. As a hybrid of the cognitive and pragmatic approaches, this model combines the theoretical foundation of cognitive psychology with the pragmatic methods of software engineering models. Finally, it allows for a more user-centered design by incorporating domain-specific considerations of cognitive processes to be performed by the learner.

In summary, the proposed model is contextualized, based on a cognitive approach, and still pragmatic enough to be practically applicable. In addition, it provides a good basis for the development of evaluation criteria, which are discussed in the following section.

EVALUATION CRITERIA FOR THE USER INTERFACE OF FL MULTIMEDIA SOFTWARE

We will now return to the original question regarding the evaluation of the user interface in FL multimedia software: "Is the design user-centered?" Based on the model proposed in the previous section, this question can be specified further by incorporating the new definition and function of the user interface. We can now ask two questions: (1) "What are the functions of the interface elements?" and (2) "How well does the user interface support the cognitive processes involved in SLA?"

This approach to the evaluation of the user interface is domain-specific and can only be used with a specific field in mind. For second language acquisition, the cognitive processes can be derived from the linguistic and pragmatic competencies and skills described earlier (see Table 1).

The process of developing domain-specific evaluation criteria for a particular software application would thus involve the following steps:

1) Identify relevant skills, competencies, and domain knowledge.
2) Identify activities that cultivate and develop these skills, competencies, and knowledge.
3) Identify the cognitive processes involved in these activities.
4) Assess the level of support for these cognitive processes provided by the application and its user interface.

Steps 1 though 3 of this process have been discussed in the previous sections of this paper. In step 4, the level of support is assessed for each of the relevant cognitive processes. In the case of reading comprehension, these processes include:

- Activating and processing prerequisite knowledge
- Decoding the linguistic surface structure of the text
- Paying attention to and selecting relevant information from the text
- Organizing the selected information into a propositional text base in short term memory
- Integrating new information with the existing knowledge into a mental model

For the assessment of the level of support for these processes one would identify the interface features supporting them and as well as the quality of the implementation. This includes an assessment of how well an interface feature supports individual learner differences, such as different cognitive or learning styles. This approach can accommodate new activities and instructional strategies and methods since it does not attempt to compile a comprehensive list of all activities known, but rather assesses whatever activities were implemented by the designers of the software. A sample evaluation form is outlined in Table 3.

Table 3 is given as an example only and is not meant to be a comprehensive list of cognitive processes or activities. Its objective is to demonstrate the approach described in this section. Each of the criteria is rated for its overall level of support, as well as for the support of individual learner differences. Similar criteria can be developed for other competencies and skills. This is, however, outside the scope of this paper. A comprehensive evaluation of a software application needs to include a number of additional sections that provide the reviewer's information, a general program description, instructional goals and objectives, the target language, the overall instructional approach or philosophy, the intended target audience, the required level of proficiency in the foreign language, technical aspects of the software, and other information specific to that particular software.

Cognitive Process	Supporting Activity	Level of Support (None Medium High)
Reading Comprehension		
Activating and processing of prerequisite knowledge	Pre-reading activities	☐ ☐ ☐ ☐ ☐
	Visual advance organizer (video)	☐ ☐ ☐ ☐ ☑
Decoding the linguistic surface structure of the text	Annotations for vocabulary items (visual, textual)	☐ ☐ ☐ ☐ ☐
	Pronunciation of vocabulary items (audio)	☐ ☐ ☐ ☐ ☐
Paying attention to and selecting relevant information	Comprehension exercises (strategically placed at logical breaks in the text)	☐ ☐ ☐ ☐ ☐
	Annotations for vocabulary items (multimodal)	☐ ☐ ☐ ☐ ☐
Organizing the selected information into a text base	Comprehension exercises with immediate feedback	☐ ☐ ☐ ☐ ☐
	Reinforcement for learning provided by multiple modes of vocabulary annotations	☐ ☐ ☐ ☐ ☐
Integrating the new information into the learner's mental model	Voice-over narration provides additional presentation mode to reinforce the visual and textual modes	☐ ☐ ☐ ☐ ☐
	Comprehension exercises require the learner to write short paragraphs	☐ ☐ ☐ ☐ ☐
	Learners write essays about the text for peers to read	☐ ☐ ☐ ☐ ☐

Table 3. Sample Evaluation Form for Reading Comprehension

The proposed approach results in evaluation criteria for software that are domain-specific and that explicitly address the level of support for individual differences in each cognitive process. In addition, this approach provides an adaptive evaluation method that can accommodate existing and future instructional methods and activities. An implementation of such an adaptive evaluation system could be done using an adaptive hypertext that modifies the list of criteria for each application, based on the activities implemented in a particular program.

CONCLUSIONS

In the preceding sections I gave an overview of existing models and approaches to user interface design and discussed their main focus and strengths. The main problem with these approaches lies in the fact that they are either very pragmatic and not based on underlying theories, or that they are theory-driven but too complex to be used in the design process. In order to derive an approach specifically targeted for SLA software, I first reviewed the linguistic and pragmatic competencies that are addressed in FL instruction and then described a new hybrid approach to interface design for FL multimedia software.

This approach combines the theoretical basis of a cognitive approach with the pragmatic methods of software engineering approaches. First and foremost, it is based on the competencies and skills to be developed and the cognitive processes underlying them. Second, it incorporates rapid prototyping, or the use of CASE tools. It is argued that a contextualized cognitive approach to interface design can lead to a more domain-specific support of cognitive processes involved in the acquisition of FL competencies and skills, and will result in a more user-centered design of the user interface. In addition, it will allow for the development of an adaptive domain-specific set of evaluation criteria based on this level of support. I applied the proposed model to the design of software for reading comprehension and for developing evaluation criteria for such software. It goes without saying that empirical research is needed to test the model's effectiveness.

While the model for the design and evaluation of the user interface proposed in this paper was demonstrated in its application to SLA, it has the potential to provide a general framework for the development of user-centered instructional software and for the development of domain-specific evaluation criteria in other disciplines.

NOTE

This paper is based on a paper presented at the Invitational Symposium on Assessing & Advancing Technology Options in Language Learning (AATOLL) at the National Foreign Language Research Center of the University of Hawai'i in February 1998 in Honolulu, HI.

ABOUT THE AUTHOR

Jan L. Plass is an Assistant Professor in the Organizational Learning and Instructional Technologies Program at the University of New Mexico. His research interests include learning with multimedia, especially L2 vocabulary acquisition and reading comprehension, interface design and constructivist learning in networked environments.

E-mail: jan@unm.edu

REFERENCES

Anglin, G. J. (Ed.). (1991). Instructional technology: Past, present, and future (2nd ed.). Englewood, CO: Libraries Unlimited, Inc.

Barnard, P. (1991). Bridging between basic theories and the artifacts of human-computer interaction. In J. M. Carroll (Ed.), *Designing interaction: Psychology of the human-computer interface* (pp. 103-127). Cambridge, MA: Cambridge University Press.

Buxton, W., & Lamb, M. R. (1983). Towards a comprehensive user interface management system. *Computer Graphics*, *17*, 35-42.

Card, S. K., Moran, T. P., & Newell, A. (1983). *The psychology of human-computer interaction*. Hillsdale, NJ: Erlbaum.

Carroll, J. M. (1991). The Kittie House Manifesto. In J. M. Carroll (Ed.), *Designing interaction: Psychology of the human-computer interface* (pp. 1-16). Cambridge, MA: Cambridge University Press.

Chun, D. M. (1994). Using computer networking to facilitate the acquisition of interactive competence. *System*, *22*(1), 17-31.

Chun, D. M. (1998). *Signal analysis software for teaching discourse intonation*. Paper presented at the Invitational Symposium: Assessing & Advancing Technology Options in Language Learning (AATOLL). University of Hawai'i, National Foreign Language Research Center, Honolulu, HI.

Chun, D. M., & Plass, J. L. (1995). Project CyberBuch: A hypermedia approach to computer-assisted language learning. *Journal of Educational Multimedia and Hypermedia, 4*(1), 95-116.

Chun, D. M., & Plass, J. L. (1996a). Effects of multimedia annotations on vocabulary acquisition. *The Modern Language Journal*, *80*, 183-198.

Chun, D. M., & Plass, J. L. (1996b). Facilitating reading comprehension with multimedia. *System*, *24*(4), 503-519.

Chun, D. M., & Plass, J. L. (1997a). Research on text comprehension with multimedia. *Language Learning & Technology*, *1*(1), 60-81. Retrieved August 8, 1997 from the World Wide Web: http://polyglot.cal.msu.edu/llt/vol1num1/chun_plass/default.html.

Chun, D. M., & Plass, J. L. (1997b). *CyberBuch*. [Computer Program, Macintosh]. New York: St. Martin's Press.

Chun, D. M., & Plass, J. L. (1998). *Ciberteca: Una Carta a Dios*. [Computer Program, Macintosh, Windows 95/NT]. New York: St. Martin's Press.

Clark, R. E., & Sugrue, B. (in press). Media selection for training. In S. Tobias & D. Fletcher (Eds.), *APA handbook on training*. Washington, DC: American Psychological Association.

Cockton, G. (1988). Generative transition networks: A new communications control abstraction. In D. M. Jones & R. Winder (Eds.), *People and computers IV* (pp. 509-525). Cambridge, MA: Cambridge University Press.

Damodaran, L., Ip, K., & Beck, M. (1988). Integrating human factor principles into structured design methodology: A case study in the U.K. civil service. In H. J. Bullinger (Ed.), *Information technology for organisational systems*. Amsterdam: Elsevier.

Dayton, T. (1991). Cultivated eclecticism as the normative approach to design. In J. Karat (Ed.), *Taking software design seriously* (pp. 21-44). Cambridge, MA: Academic Press.

Ehsani, R., & Knodt, E. (1998). *Speech recognition in computer aided language learning*. Paper presented at the Invitational Symposium: Assessing & Advancing Technology Options in Language Learning (AATOLL). University of Hawai'i, National Foreign Language Research Center, Honolulu, HI.

Eskenazi, M. (1998). *Issues in the use of speech recognition for foreign language tutors.* Paper presented at the Invitational Symposium: Assessing & Advancing Technology Options in Language Learning (AATOLL). University of Hawai'i, National Foreign Language Research Center, Honolulu, HI.

Furnes, T. A., & Barfield, W. (Eds.). (1995). *Virtual environments and advanced interface design*. New York: Oxford University Press.

Gardiner, M. M., & Christie, B. (Eds.). (1991). *Applying cognitive psychology to user-interface design.* Chichster, New York: John Wiley.

Kieras, D. E., & Polson, P. G. (1985). An approach to the formal analysis of user complexity. *International Journal for Man-Machine Studies*, *22*, 365-394.

Landauer, T. K. (1991). Let's get real: A position paper on the role of cognitive psychology in the design of humanly useful and usable systems. In J. M. Carroll (Ed.), *Designing interaction: Psychology of the human-computer interface* (pp. 60-73). Cambridge, MA: Cambridge University Press.

Laurel, B. (Ed.). (1990). *The art of human-computer interface design*. Reading, MA: Addison-Wesley.

Levin, J. R., Anglin, G. J., & Cary, R. N. (1987). On empirical functions of pictures in prose. In D. H. Willows, & H. A. Houghton (Eds.), *The psychology of illustration*, Vol. 1, pp. 51-85. New York: Springer.

Luperfoy, S. (1998). *Tutoring versus training in spoken dialogue instructional systems.* Paper presented at the Invitational Symposium on Assessing & Advancing Technology Options in Language Learning (AATOLL). University of Hawai'i, National Foreign Language Research Center, Honolulu, HI.

Marchionini, G. (1991). *Psychological dimensions of user-computer interfaces*. Syracuse University. (ERIC Document Reproduction Service No. ED 337 203)

Mayer, R. E. (1997). Multimedia learning: Are we asking the right questions? *Educational Psychologist*, *32*(1), 1-19.

Norman, D. (1990). *The design of everyday things*. New York: Doubleday Currency.

Norman, D. A. (1987). Design principles for human-computer interfaces. In R. M. Baeker & W. A. S. Buxton (Eds.), *Readings in human-computer interaction: A multidisciplinary approach* (pp. 492-501). San Francisco: Morgan Kaufman.

Park, I., & Hannafin, M. J. (1993). Empirically-based guidelines for the design of interactive multimedia. *Educational Technology Research and Development*, *41*, 63-85.

Plass, J. L., Chun, D. M., Mayer, R. E., & Leutner, D. (1998). Supporting visual and verbal learning preferences in a second language multimedia learning environment. *Journal of Educational Psychology*, *90*(1), 25-36.

Price, P., & Rypa, M. (1998). *Speech technology for language education.* Paper presented at the Invitational Symposium: Assessing & Advancing Technology Options in Language Learning (AATOLL). University of Hawai'i, National Foreign Language Research Center, Honolulu, HI.

Ravden, S. J., & Johnson, G. I. (1989). *Evaluating usability of human-computer interfaces: A practical method*. Chichester, UK: Ellis Horwood.

Rubinstein, R., & Hersh, H. (1984). *The human factor: Designing computer systems for people*. San Francisco: Digital Press.

Shneiderman, B. (1993). *Designing the user interface: Strategies for effective human-computer interaction* (2nd ed.). Reading, MA: Addison-Wesley.

Smith, P. L., & Ragan, T. J. (1993). *Instructional design*. New York: Merrill.

Sutcliffe, A. (1988). Some experiences in integrating specification of human-computer interaction with a structured system development method. In D. M. Jones, & R. Winder (Eds.), *People and computers IV* (pp. 146-160). Cambridge, MA: Cambridge University Press.

Sutcliffe, A. (1989). Task analysis, systems analysis and design: Symbiosis or synthesis. *Interacting with Computers*, *1*, 6-12.

van Dijk, T., & Kintsch, W. (1983). *Strategies of discourse comprehension*. New York: Academic Press.

Wallace, M. D., & Anderson, T. J. (1993). Approaches to interface design. *Interacting with Computers*, *5*(3), 259-278.

Warschauer, M. (Ed.). (1997). *Virtual connections: Online activities & projects for networking language learners* (Technical Report #8). Honolulu, HI: University of Hawai'i Press.

Wasserman, A. I. (1985). Extending state transition diagrams for the specification of human-computer interaction. *IEEE Trans Software Engineering*, *11*, 699-713.

Waterworth, J. A. (1992). *Multimedia interaction with computers: Human factors issues*. Chichester, England: Ellis Horwood.

Winograd, T. (Ed.). (1992). *Bringing design to software*. Reading, MA: ACM Press/Addison-Wesley.

Wroblewski, D. A. (1991). The construction of human-computer interfaces considered as a craft. In J. Karat (Ed.), *Taking software design seriously* (pp. 1-19). Cambridge, MA: Academic Press.