DOCUMENT RESUME

ED 284 453 FL 016 859

AUTHOR Hubbard, Philip L.
TITLE Language Teaching Approaches, the Evaluation of CALL Software, and Design Implications.
PUB DATE 82
NOTE 29p.; In: Smith, William Flint, Ed. Modern Media in Foreign Language Education: Theory and Implementation. The ACTFL Foreign Language Education Series; see FL 016 850.
PUB TYPE Information Analyses (070)

EDRS PRICE MF01 Plus Postage. PC Not Available from EDRS.
DESCRIPTORS *Computer Assisted Instruction; *Computer Software; Educational Technology; *Evaluation Criteria; *Evaluation Methods; Language Processing; Learning Processes; Linguistic Theory; Program Design; *Second Language Instruction; Teaching Methods; Theory Practice Relationship

ABSTRACT

A computer-assisted language learning (CALL) software evaluation form that looks directly at important aspects of language teaching approaches is presented. Examples of typical evaluation checklists are described to illustrate how they fail to address the appropriate questions for language teaching software. Three major categories of language teaching approaches (behaviorist, explicit learning, and acquisition) are defined and discussed as to how the principles can be applied to software evaluation. Ways in which to evaluate software for its potential use in teaching strategies for learning are discussed, and the relationship between software, approach, and syllabus is considered. A procedure is outlined for using the form to evaluate CALL materials, and implications of these concepts for software design are suggested. (CB)

Language Teaching Approaches, the Evaluation of CALL Software, and Design Implications

Philip L. Hubbard
Stanford University

Introduction

Evaluating software for computer-assisted language learning (CALL) is a formidable task, even for those who are relatively familiar with the field. Whether one carries out an evaluation informally for a local institution or writes more formally for published review, there are at least five parameters of the judgmental process that make the evaluation of courseware more difficult and challenging than that of conventional textbooks.

First, there is the sheer novelty of this infant field of instructional technology to contend with. Reviewers of textual materials generally have had many years of experience working with print madia as student and teacher. In contrast, software evaluators may be only recently computer literate, and the field is new enough that few can boast more than a few years of experience using computer materials in their classes; fewer still have ever experienced CALL from the students' perspective.

Second, there is usually no way to "skim" through the software as with a textbook and other print materials. With few exceptions, the evaluator must proceed relatively lockstep once a choice is made from the main

Philip L. Hubbard (Ph.D., University of California, San Diego) is Lecturer in Linguistics and English for Foreign Students and Coordinator for Summer Programs in ESL at Stanford, where he also teaches English for foreign students and EFL/ESL methodology. He is the founder of the Clearinghouse for ELS Public Domain Software at the Ohio University (Athens), and a member of TESOL and CALICO.

menu. This restriction makes it very difficult to obtain an overview of the lesson format and content as quickly as is possible with a textbook.

Third, the relative complexity in the placement of lesson components on a disk makes the review process laborious. The hierarchical structuring and branching routines typically found in some of the more complex programs may define a large number of possible paths through a given lesson, which makes it difficult to view all of the material that a student might access while using the disk.

Fourth, there are visual and auditory dimensions in computer software missing in print material and that require evaluation. In the visual dimension, while both a textbook and computer software may include text alone, text plus pictures, or diagrams and charts, only the computerized lesson can animate graphics and highlight aspects of the text in a dynamic way. And speech synthesis aside, there are the congratulatory and critical buzzes, beeps, and electronic melodies that require a judgment whether they actually enhance or detract from the lesson.

Finally, there are interactional aspects to consider. To what extent does the student control the lesson or vice versa? How "intelligently" does the program evaluate the student's input? How does the computer respond after having made that evaluation? A program—whether drill and practice, tutorial, or simulation—in a real sense acts as teacher; the evaluator must determine to what extent that "teacher" is able to aid the student in learning the target language.

The relative newness of CALL software has meant that most guidelines or checklists for software evaluation concentrate largely on the parameters mentioned above, along with the equally vital questions about what hardware is required to run a given piece of software. As useful as such evaluation forms may be, however, they typically fail to address the crucial questions related to *approach*, the essential foundation of all language instructional material. While the variables mentioned above are important to evaluate, the bottom line must be "How is this piece of software going to improve a student's proficiency in the target language?"

This chapter discusses a supplementary evalution form developed to aid in answering this question by looking directly at important aspects of language-teaching approaches, where "approach" is taken to mean the set of underlying principles that outline a set of conditions for successful language learning and that, in turn, often follow from a theory of language acquisition and provide the foundation for specific classroom methods and techniques (Anthony, 1). Thus, the first section in this chapter shows how a typical checklist fails to address the appropriate questions for language-teaching software. The second section defines three major categories of language-teaching approaches and suggests that the principles they encompass can be applied to software evaluation. The third section discusses the evaluation of software for its potential in teaching the student strategies for learning, and the fourth will consider the relationship

between software, approach, and syllabus. The final section introduces the supplementary evaluation form and outlines a procedure for using it to evaluate CALL materials. The chapter concludes with some implications of these concepts for software design.

Problems with Existing Checklists

There are a number of published software evaluation checklists, but many, such as the MicroSIFT form produced by the International Council for Computers in Education (Marler, 31), have been designed for use with CAI materials in any field. Using them to review foreign language materials assumes that learning a second language is essentially the same as other types of learning when, in fact, it is not. However strongly psychologists, linguists, and language teachers may disagree on other issues, the overwhelming majority support the view that learning a second language differs in significant ways from learning anything else.

In addition to such general-use forms and questionnaires, other checklists that have been developed specifically for second and foreign language materials (Strei, 42, Decoo, 17, Hope et al., 23, and Curtin and Shinall, Chapter 10 in this volume) similarly do not focus on the relative fit of the software to instructional approach and require supplementation to extract meaningful judgments whether the materials are likely to accomplish their advertised ends. The checklist formulated by CALICO (currently under revision) is a case in point, as can be seen in Figure 1 (an abridgment by this author of the original in which contents irrelevant to language-teaching approach considerations have been omitted and the items renumbered for internal consistency).

I. General Questions
   1. Is the courseware intended to be teacher dependent or teacher independent?
   2. Are the exercises mechanical, meaningful, or communicative?

II. Pedagogical Considerations
   1. Are the courseware contents substantively correct?
   2. Are the explanations complete and adequate in number?
   3. Are there adequate examples?
   4. Are concepts presented well?
   5. Can the courseware be used to introduce material as well as reinforce it?
   6. Are there adequate application activities?
   7. Are there adequate evaluation activities?
   8. Are all instructional units of approximately the same length?
   9. Are all the instructional units at approximately the same level of difficulty?
   10. Is the purpose of the package well defined?

11. Does the package achieve its defined purpose?
12. Is the package motivational?
13. Is the level of difficulty found in the courseware appropriate to the target audience?
14. Is the courseware suitable to students with a wide range of ability levels?
15. Does the level of difficulty vary according to the demonstrated ability level of the student?

III. Adaptability to Computer Medium
1. Is the instructional program interactive?
2. Is it clearly individualized?
3. Are branching or help options provided for students who need remedial attention?
4. Does the student have more than one chance to answer the question correctly?
5. Are opportunities provided for student feedback?
6. Can the student return to the start at any time?
7. Can the student exit at any time?
8. Does the student control the rate of presentation?

Figure 1. Abridged Version of the CALICO Software Evaluation Form

While lack of space prevents looking at each item in Figure 1, a significant weakness throughout is the omission of the word *language*, which detracts significantly from the face validity of the instrument in evaluating language materials. The CALICO form thus is just as suitable for evaluating CAI materials in chemistry or mathematics as it is for judging language-teaching courseware. A second problem with the CALICO checklist is a bias toward what can be called an explicit learning approach, considered by some to be the dominant form of second-language teaching at both the high school and university levels, although perhaps not the most effective.

In summary, there are shortcomings in the majority of existing checklists for evaluating CALL software. This article will suggest guidelines to overcome these limitations, taking as a basis for intelligent evaluation of CALL courseware the need for the evaluator to have a clear understanding of the *approach* underlying the curriculum and the syllabus for which it is intended.

Three Categories of Approach

Over the years, a number of distinct approaches to second-language teaching have guided classroom efforts and materials to help students learn. Although the boundaries that distinguish them may not always be clear,

three major categories of approach can be identified for discussion purposes. Unlike the case of a specific approach, however, none should be taken as corresponding directly to any particular theory; rather, they are simply convenient labels for discussing distinguishable trends in second-language learning and teaching. Some of the more recently proposed theoretical models of second-language learning—Krashen (26), Bialystok (6,7) Strevens (43), Swain (44)—find representation under all the headings, while others—stimulus-response theories underlying behaviorist approaches (e.g., Brooks, 8)—are more clearly limited to a single descriptor. Nevertheless, these three categories reflect useful distinctions for materials development and CALL software evaluation, since they reflect major principled components of specific theories and models of second-language acquisition (see also Chapter 7 by Doughty in this volume).

Behaviorist Approaches

A number of authors (Dalgish, 16, Ariew, 2, Underwood, 46, Baker, 4) have noted that much of current CALL software seems to be based on a stimulus-response theory of language learning. Whether one agrees with this type of approach or not, the characteristics that distinguish behaviorism from other approaches are worth noting so that they can be recognized when they appear in CALL courseware.

Historically, behaviorist approaches to language learning are based on the principle that a response, linguistic or otherwise, is learned behavior resulting from associating that response with a given stimulus. Through positive reinforcement for correct behavior and negative reinforcement for incorrect behavior, these responses become overlearned until they are automatic. The main difference between learning a language under a behaviorist approach and learning mathematics, for example, is that most of the methods derived from it (such as audiolingualism) also place heavy stress on learning about the second-language culture.

Larsen-Freeman (28) lists a number of principles underlying the audiolingual method. Considering just those most relevant to the computer medium, CALL software will be representative of a behaviorist approach to language teaching to the extent that it does the following:

1. presents vocabulary and structure appropriate to the learner's level
2. maintains the learner's attention to task
3. does not accept errors as correct answer
4. requires the learner to input the correct answer before proceeding
5. provides the learner with positive feedback for correct answers
6. provides sufficient material for mastery and overlearning to occur
7. reinforces patterns and vocabulary presented in a lesson

8. presents grammar rules or patterns inductively with no attempt at teaching explicit formulations of them

Explicit Learning Approaches

Explicit learning approaches, including the so-called cognitive approaches in second-language teaching trace their heritage to the grammar and grammar translation methods of the Middle Ages and Renaissance. The central notion of explicit learning is that the target language can be learned through a conscious knowledge of the meanings of its words and the rules of its grammar. Currently, there are several theoretical positions taken with respect to explicit learning. Some researchers (Ellis, 19; Bialystok, 6,7) maintain that explicit learning and practice of grammar rules can lead to the automatic processing necessary for fluent conversation in the target language. Others, including Krashen (26), maintain that explicit learning cannot lead to automatic processing and that conscious linguistic knowledge can only be utilized by the Monitor, a cognitive linguistic device that applies rules consciously in speech production and edits the output at a significantly slower rate than the automatic processing system.

A somewhat different position is taken by McLaughlin et al. (32), who propose an information-processing approach that distinguishes controlled from automatic processing but assumes that either can be conscious or unconscious. In short, McLaughlin et al. allow for explicit learning to become automatic under certain conditions. Even within Krashen's Monitor Theory, however, it is acknowledged that explicit learning can be utilized when time is not a significant factor, as in writing or intensive reading. Further support for explicit learning approaches in teaching comes from a review and reanalysis by Long (29) of data from twelve experiments in language learning. Long concluded that explicit instruction at least facilitates the development of language proficiency and is more effective than language exposure alone.

If explicit learning is aimed at both consciously learned rules and their appropriate application in comprehending and producing the target language, it is desirable for the learner to have access to as much relevant information about the rules as possible. Thus, even in a standard practice exercise, whether mechanical or meaningful, it is helpful for students to have the option of requesting assistance in the form of hints to lead them to a correct response as well as answers to questions like "Why is my answer wrong?" and "Why is this answer right?" Few teachers in a typical classroom situation fail to respond to such questions, yet many current computer programs presumably aimed at explicit learning do not make this kind of assistance available. A related issue is the importance of accepting a range of appropriate answers for certain exercise items. If the

goal of the exercise is to move the student in the direction of native-speaker competence, then any learner language a native speaker would accept as correct in a given context also should be considered correct. Teachers typically accept a range of answers as correct in classroom or homework exercises (often accompanied by a comment about minor differences in meaning or register); not many CALL programs, however, are designed to accommodate this need.

Providing drill-and-practice material in explicit learning approaches that is meaningful, contextualized, and interesting to the students is a recent trend that parallels developments in communicative approaches (Madsen and Bowen, 30). In much the same vein, Oller (36, p. 49) advocates paying greater attention to "story-writing" principles in ESL teaching, suggesting that while teachers should maintain a commitment to the explicit learning of grammar, they should also look at the pragmatic factors that affect the meaningfulness and comprehensibility of discourses used for instructional purposes. Although the focus in explicit learning materials is still more on form than on language use, there are compelling arguments in support of this trend. An exercise that is meaningful requires the students to integrate grammatical form and lexical content, leading to a deeper level of psycholinguistic processing. An exercise that is appropriately contextualized (e.g., a coherent paragraph or a series of sentences relating to a common theme, a picture, etc.) will lead the students toward the pragmatic competence necessary to extend the rules or vocabulary items correctly to novel instances. Finally, material that is interesting is more likely to hold students' attention and motivate them to complete the exercise successfully and attempt others of a similar type.

A final point within the explicit learning paradigm that relates exclusively to CALL materials involves student control of the software. Higgins (21) has argued that student-controlled software will be far more effective than lockstep program-controlled learning. Research by Stevens (41) supports this view, although the optimal degree of student control will no doubt vary with the student's level and the task. A study by Hubbard et al. (25) also bears on this issue by pointing out that if students are given a significant amount of control without appropriate training in how to use the software, they may fail to utilize the powerful options available to them.

Translating the above considerations into evaluation criteria for CALL, software will be representative of explicit learning approaches to the extent that it does the following:

1. introduces or reviews grammar rules and word meanings in an understandable, learnable, and reasonably accurate form
2. provides effective practice so that (a) novel target-language input can be readily understood, and (b) the learner's understanding of rules leads to the production of grammatically acceptable spoken or written target-language discourse in novel situations

3. gives meaningful rather than mechanical practice
4. gives practice contextualized in a coherent discourse larger than a single sentence
5. provides hints of various types to help lead students to acceptable answers
6. accepts alternative correct answers within the given context
7. provides the student with explanation of correct answers
8. anticipates incorrect or inappropriate answers and explains why such answers are incorrect or inappropriate
9. maintains the student's interest throughout the exercise
10. allows an appropriate degree of student control

In short, software that follows the principles of explicit learning as outlined above should be essentially tutorial in its approach and based on paragraph-length, episodically related examples of realistic language rather than isolated sentences. In programming the lesson, the author must anticipate the kinds of information that would most help the student learn and retain the material and provide that information in a form that is accessible and understandable. Drill-and-practice software, especially the type whose exercises can be processed in a superficial way, is simply less likely to lead to the learning and retention of explicit rules and vocabulary than is software that requires a deeper level of cognitive processing.

Acquisition Approaches

While the learning/acquisition distinction has only been popularized recently through the work of Krashen (26) and Dulay, Burt, and Krashen (18), the idea that languages can be learned effectively without formal study of language structure and vocabulary has been with the field of second-language learning for some time. To some extent, the Direct Method and other implicit learning approaches are based on this idea. The current revival of the notion of natural language learning can be traced to Newmark (34) and Newmark and Reibel (35), who suggested that languages can be learned in a natural environment that holds meaningful and understandable "chunks" of language provided that the learner simply pays attention to the language as it is being used. Krashen (26) later refined these ideas into a theory of second-language learning and acquisition involving (1) the Monitor, mentioned in the previous section, for explicit learning and processing, (2) an Organizer, or Language Acquisition Device, and (3) an Affective Filter (Dulay, Burt, and Krashen, 18). The Organizer and Affective Filter are central to the acquisition portion of the overall model, as they are involved in the learner's unconscious analysis of strings of speech in the processing of speech input, the production of output, and the synthesis of new rules.

Krashen's model, though far from being universally accepted, has

exerted a profound influence on second-language acquisition research in the past decade. It has also provided a theoretical foundation for a class of communicatively oriented approaches (Terrell, 45; Krashen and Terrell, 27; Winitz, 49). There is, in fact, a significant overlap between acquisition approaches and communicative approaches in general, even though the two are not synonymous. Specifically, it is possible to include communicative methodology within the umbrella of an explicit learning approach. As Brumfit (10, p. 37) has pointed out, however, a number of second-language acquisition researchers and language-teaching methodologists have assumed a dichotomy roughly parallel to the learning/acquisition division in Krashen's theory and, while they vary somewhat with respect to the role explicit learning might play in a communicative approach, the importance of acquisition in developing communicative competence is generally supported. For this reason, in the discussion that follows communicative approaches are subsumed under "acquisition." (See Doughty, Chapter 7 in this volume, for a further discussion of the acquisition/learning distinction.)

The first requirement for acquisition in Krashen's model is that the input (the language directed at the learner) be comprehensible, i.e., that its meaning can be determined by the learner from the communicative context and the linguistic information in the learner's acquired system at that point in the acquisition process. Given some linguistic input, the Affective Filter determines how much of this input makes its way to the Organizer for processing. Two significant variables in the Affective Filter are the learner's attitude and motivation. A positive self-image, a low level of anxiety, and a strong desire to learn the language (and engage in the communicative tasks necessary for acquisition) are among the criteria for keeping the Affective Filter at a low level, thus allowing maximum linguistic input into the Organizer. Finally, the learning environment and the types of learning tasks are assumed to have a significant effect on the Affective Filter.

Krashen's Organizer is reminiscent of the Language Acquisition Device that Chomsky (13) hypothesized for first-language acquisition. Input that has not been filtered out, or *intake* as Krashen calls it, is unconsciously analyzed by the Organizer, which also formulates rules (again unconsciously). While the internal operation of the Organizer remains a mystery, it is hypothesized to operate most effectively on linguistic structures that are just beyond the level already attained in the acquired grammar. Because of the implied developmental sequence in the acquisition of grammatical rules, it is also generally assumed that overt error correction will have no real value in acquisition and, in fact, may be detrimental, because correcting learner errors leads to anxiety and raises affective barriers.

Krashen's model focuses on the learner's linguistic input and does not require any production explicitly in order for acquisition of linguistic and

communicative competence to occur. However, acquisition-based approaches generally put some emphasis on developing the learner's ability to produce as well as to understand novel utterances. Thus, a final point about acquisition approaches is that they allow the learner practice in producing comprehensible *output* in addition to processing input.

Although Krashen's model has had little influence on the development of CALL software, it is not the case that CALL specialists have ignored acquisition approaches altogether. Higgins and Johns (22), Dalgish (16), and Baltra (5) discuss the use of games and simulations as communicative activities and stress the value of learner–learner conversation in front of the screen. Having pairs and even larger groups of learners working together at a single screen allows the computer material to act as a catalyst to promote real communicative interaction between the participants. Underwood (46), another advocate of communicative CALL, discusses at length the need for developing software that is consistent with current linguistic and language-learning theory. His thirteen premises for communicative CALL have much in common with the evaluation criteria that are presented below. Unlike these evaluation criteria, however, many of Underwood's premises focus on what communicative CALL is *not*: "Communicative CALL will never try to do anything a book could do just as well" (p. 54), while some of his other premises are more consultatory: "Communicative CALL will use the target language exclusively" (p. 53).

Looking at the above factors in terms of computer courseware, and following Krashen's theoretical model with the addition of some insights from communicative methods, CALL software will be representative of an acquisition-oriented approach to the extent that it does the following:

1. provides meaningful communicative interaction between the learner and the computer
2. provides comprehensible input at a level just beyond that currently acquired by the learner
3. promotes a positive self-image in the learner
4. motivates the learner to use the software
5. motivates the learner to learn the language
6. provides a challenge but does not produce frustration or anxiety
7. does not include overt error correction
8. allows the learner the opportunity to produce comprehensible output
9. acts effectively as a catalyst to promote learner–learner interaction in the target language

Note that "meaningful communicative interaction" (number 1 above) does not necessarily imply that both the learner and the computer use language in an exchange. It is possible for the computer to react to input nonlinguistically (e.g., by moving text or graphics in response to the learner's command) and for the learner to react to the computer's target-language output nonlinguistically (e.g., by moving the cursor or a graphics

figure with pointing devices, such as the cursor keys or a mouse, in response to the computer's command). In fact, the latter type of interaction is typical of beginners in delayed production approaches and methods such as the Natural Approach (Terrell, 45; Krashen and Terrell, 27), Total Physical Response (Asher, 3), and the Comprehension Approach (Winitz, 49).

This section has focused on three categories of approaches—behaviorist, explicit learning, and acquisition—which were defined broadly in the interest of comprehensiveness. These broad categories of definition are not meant to imply, however, that all approaches will fit directly under one heading; rather, a given approach may not share all of the principles of its closest superordinate category and may include other significant principles not mentioned above (such as whether the native language may be used at all in instruction). In practice, too, the individual teacher may quite consciously vary the approach, based on perceived student needs. For example, in second or foreign language classes or programs whose goals include grammatical accuracy as well as communicative competence (as is often the case with languages taught for academic or professional purposes), it is not uncommon to see a preference for acquisition approaches in conversation/discussion components and explicit learning approaches in reading/writing/grammar components. Often, there is variation based on the learner's level as well, with acquisition approaches favored for beginning levels and explicit learning approaches favored for more advanced ones. Nevertheless, the parameters listed above for these categories are useful aids in determining the language-teaching approach underlying given CALL software and, in turn, provide a means for judging a fundamental question in software evaluation: How well does the teaching approach manifested in the software match that of the teacher whose students will be using it?

Learner Strategy Orientation

An additional area of consideration in judging the probable effectiveness of foreign language software is the degree to which the materials may directly or indirectly promote the use of particular strategies in the learner, that is, procedures for learning, acquiring, or using the target language more easily and effectively. The concept of learner strategies is not limited directly to any particular category of approaches. In fact, it is possible for the learner to be taught or to develop strategies compatible with approaches in any of the three major categories mentioned above.

The idea of using strategies to enhance learning is not new. Learner strategies have received increasing attention from both researchers in second-language learning and materials developers in the past decade as

part of the general trend toward focusing on the language learner, and several taxonomies have been proposed to describe them. Most, in one way or another, distinguish direct or cognitive learner strategies (those involving deliberate manipulation of material to enhance learning or retention) from indirect or metacognitive ones (those involving self-monitoring, self-assessment, and goal setting—Oxford-Carpenter, 37, p.1). Wenden (47) has added communication strategies (those used to facilitate information exchange) and global practice strategies (those that lead the learner to utilize the environment effectively for target language practice). Rather than attaching itself to any particular taxonomy, however, the discussion that follows will look at learner strategies from the viewpoint of some general principles that are particularly appropriate for the evaluation of strategy-oriented software.

In a sense, any structured or semistructured exercise, from pattern practice to open-ended role play, represents a strategy on the part of the teacher for aiding the student in gaining proficiency in the target language. But these are really *teaching* strategies. A *learner*-strategy orientation, on the other hand, involves focusing more on those strategies that the learner may come to employ consciously and control independently. For example, in teaching vocabulary with a learner-strategy orientation, the focus is not on learning individual lexical items; instead, the teacher introduces and provides meaningful practice in strategies for guessing the meaning of an unknown word (attending to context, deducing the part of speech, decomposing it into stem and derivational morphemes, etc.). In the first instance, for example, the student can be taught the strategy of looking ahead to the words that follow the unknown word for clues to its meaning, rather than simply ignoring it or stopping and reaching for the dictionary as soon as it is encountered. As this example illustrates, the teaching of strategies helps move the learner from being a passive recipient of linguistic input to becoming a more conscious and autonomous processor; in short, learner strategies shift much of the responsibility for successful learning from the teacher to the student (Wenden, 47, p. 5).

It is significant for CALL that many of the strategies that are useful in second-language learning are utilized by native speakers as well. This is because many learner strategies are aimed primarily at improving the individual's performance, rather than achieving linguistic competence. Thus, CALL software developed for native speakers to enhance their strategies for learning, particularly in the areas of reading and writing, may be quite helpful for advanced learners of a target language.

As with other strategy-oriented material, the focus for CALL software should be on strategies that can be learned consciously or induced in the learner, rather than on those that seem to be a universal by-product of second-language learning, e.g., overgeneralization and avoidance as discussed by Brown (9). Strategies to be taught may range from the very

general, e.g., the fourteen major ones for "the good language learner" discussed by Rubin and Thompson (38)—learn to tolerate ambiguity, let context help you, learn formalized routines, etc.—to the very detailed and skill-specific, e.g., skimming a passage by reading the first sentence of each paragraph to gain a general idea of content (Grellet, 20).

There are several types of strategies that seem particularly well-suited to being introduced and practiced on the computer. In reading, for example, psycholinguistic research has pointed to the importance of top-down processing strategies, using information from skimming a passage, from its title and introduction, or from the reader's background knowledge of the subject to build an anticipation schema for its rhetorical structure and content (Coady, 14; Carrell and Eisterhold, 12). Experiments reported in Carrell (11) provide evidence that ESL learners even at the advanced level make less use of such strategies than do typical native English readers, and it is plausible to assume that this may be true of second and foreign language reading in general. In writing, there are production strategies such as writing dialogs, brainstorming, list making, and flexible outlining that many second-language learners are either unaware of or ignore (Spack, 40; Hubbard, 24). At the level of rhetorical structure, Scarcella (39) has shown that the reader orientation strategies of advanced ESL writers differ significantly from those of native writers, which suggests that strategies learned in the first language may not transfer to or be appropriate in a second.

Unlike the behaviorist, explicit learning, or acquisition approaches, there are no specific learning theories or models associated with learner strategies. Consequently, much of what follows is based on the writer's views of how to teach learner strategies effectively.

A number of factors must be considered in producing materials to teach learner strategies. A particular strategy will be effective to the degree that it fits both the learner's needs and his or her preferred learning style. If the strategy is being taught explicitly, as is generally the case, it must be presented in a comprehensible way and be accompanied by an explanation of the principles underlying it—without this, the learner is unlikely to be convinced of its value. Reinforcement tasks, then, should be designed so that they are accomplished more efficiently if a given strategy is used appropriately. In many cases, a variety of related strategies may be presented together with explanations of their respective strengths and weaknesses. In this instance the accompanying exercises will give the learner the opportunity to experiment and discover which strategies work best under specific conditions.

Based on the preceding considerations, CALL software will effectively promote the learning and use of learner strategies to the extent that it does the following:

1. introduces the learner to strategies that are useful and immediately usable
2. introduces the learner to strategies appropriate to the learner's level

3. explains the value of the strategies
4. provides meaningful practice in the use of the strategies
5. presents practice material in such a way that the task is more easily or successfully accomplished if the appropriate strategy or strategies are used
6. provides, when possible, a variety of strategies (or of techniques for utilizing a given strategy) for a given type of task suited to a range of learning styles
7. provides feedback on which strategies might have worked best for given tasks after the learner has attempted them

As research continues in the area of learner strategies, it is likely that more explicit methods and techniques will be developed to teach and to judge their relative effectiveness. In the meantime, the parameters mentioned above provide a usable evaluation metric for determining the potential effectiveness of strategy-oriented software.

The Syllabus and CALL Materials

The software evaluator, particularly one making decisions for his or her class or program rather than for published review, must determine to what extent the orientation and content of the software is compatible with the orientation, content, and sequencing of an institution-specific syllabus. Some of the parameters that are useful to consider in answering that question are discussed below for four major syllabus types—structural, situational, notional–functional, and content-centered—recognized generally as typical in second or foreign language teaching.

Structural Syllabus

The focus of a structural syllabus is the grammar of the target language. Within a behaviorist or explicit learning approach, the grammar patterns or rules of the language are typically sequenced along continua of presumed ease of learning (easiest structures first) and to a lesser extent relative frequency (more common structures first). In either case, the structural focus of CALL software to be used as an integral part of a course must be at a level that is consistent with that prescribed by the syllabus. If the software is used on a voluntary basis by the students, then it may be appropriate to have material at a lower level for remedial purposes.

A learner-strategy orientation in CALL can be incorporated into a structural syllabus to the extent that the software is designed to introduce and practice general procedures for learning and using grammatical constructions. As reading is often a component of an explicit learning

approach in a structural syllabus, software that focuses on appropriate reading strategies can also fit into this combination.

The structural syllabus per se is not really compatible with an acquisition approach, although it is possible to have genuine communicative exercises, either on the computer itself or involving the learners in conversation with the computer acting as a catalyst, which lend themselves naturally to promoting the use of a particular grammatical construction (e.g., giving instructions would promote the use of imperative forms).

Situational Syllabus

In a situational syllabus, the focus is on introducing the learners to situations they are likely to encounter in using the target language (ordering a meal, cashing a check) and helping them to build frames of reference for communicating appropriately. In practice, a situational syllabus is often integrated with a structural or notional–functional syllabus. However, the central focus in a true situational syllabus is on the relevant vocabulary and the interchanges between the participants within a prototypical context, ideally leading the learner to an understanding of the expectations, both linguistic and nonlinguistic, of target language speakers. Regardless of the approach, CALL software being reviewed for use within a situational syllabus should demonstrate content and situations which the learner is likely to encounter in real life; ideally, it should be sequenced to reinforce or complement the situations the learner is exposed to in the text and classroom.

When a behaviorist approach is applied to a situational syllabus, the goal is presumably to drill the learner in prototypical situations and variants with a similar communicative structure. CALL software that involves this type of drill, particularly if it allows for the repetition necessary for habit formation, would be consistent with this combination.

If an explicit learning approach is linked to a situational syllabus, CALL software consistent with the combination would need to provide lists of vocabulary to be learned, examples of appropriate interchanges, and a discussion of what the structural and sociolinguistic elements of the interchanges are, along with practice in initiating the conversation and responding appropriately. Bearing in mind earlier comments about the tutorial nature of good explicit-learning software, hints and explanatory feedback should also be provided. An appropriately programmed interactive videodisc containing a common situation, such as learning the language and behavioral protocols of going to a restaurant, would be an example of explicit learning within a situational syllabus.

An acquisition approach can be maintained in a situational syllabus by using CALL software whose content provides comprehensible input in the given situations and that allows the learner to interact with the computer

and/or other learners. Simulations of common situations, for example, could provide both communicative language practice and information about target culture expectations, e.g., getting from the airport to a specific location in a strange city.

Learner-strategy software within a situational approach is also a possibility. In this case, the focus of the software is on strategies for acquiring information about the given situation by observing or asking native speakers for assistance, and on strategies for identifying and repairing miscommunication in an acceptable manner according to the target culture standards of behavior.

Notional-Functional Syllabus

Strictly speaking, a notional syllabus divides the target language into semantic categories (greeting, apology, anger, etc.) while a functional syllabus focuses on language use (asking, expressing feelings, responding, etc.). In practice, they are generally combined into a syllabus built around using language within particular semantic categories—asking for assistance, expressing anger, responding to an apology, etc. (Wilkins, 48). While it is possible to realize a behaviorist or explicit learning approach within a notional/functional syllabus, the focus on real language use that lies at the root of this syllabus orientation makes it much more compatible with acquisition approaches.

CALL software that allows for real communicative interaction between the learner and computer or between two or more learners in front of the computer will be in line with an acquisition approach in a notional-functional syllabus to the extent that it meets the criteria for acquisition-oriented software in general and provides practice in using the notions/functions called for by the syllabus at the level it is being used. Software that reviews notions/functions from earlier lessons can provide acceptable material toward this end as well, as long as the lessons remain challenging enough to engage the learners' interest. However, a reviewer should be wary of software that might assume that the learners have mastered a particular notion/function (e.g., making polite requests) that in fact they do not control, even if the grammar and vocabulary are familiar to them.

Learner strategies relate well to a notional-functional syllabus, since some of the notions/functions typically taught (asking for clarification, restating, etc.) can be viewed as communicative strategies as well. The type of CALL software that would be relevant is similar to that mentioned above for the situational syllabus, i.e., that which focuses on strategies that aid the learners in acquiring information, comprehending it, and making themselves understood.

Content-Centered Syllabus

In a true content-centered syllabus, language is not the focus; rather, it is the tool the student learns to manipulate to get at information about the subject matter of the course (Mohan, 33). To a limited extent, content-centered courses have been around for years in the form of foreign language classes whose material relates to teaching about the target culture. Typically, however, such courses have tended to have structural sequencing as the overriding consideration; that is, content-centered courses simply inject content into a structural syllabus. More recently, building on the model of bilingual transition programs where certain subjects are taught in the child's second language, content-centered courses focusing on other topics have appeared in second and foreign language teaching (Curtain, 15), particularly in English for academic programs and "language" for business classes in the United States. Structural, situational, and notional-functional aspects of the language may still be considered within this more recent view of content-centered language teaching, but they are of secondary importance.

Like the notional-functional syllabus, the content-centered syllabus is founded on the assumption that meaningful, natural use of language in a realistic communicative setting will lead to increased proficiency in the target language (Mohan 33, p. 1). Thus, the content-centered syllabus is not really compatible with the drill-and-practice focus of behaviorist approaches or the language-analysis orientation of explicit learning approaches. It is, however, quite compatible with acquisition approaches and a learner-strategy orientation.

Within an acquisition approach, there are three important considerations in evaluating content-centered CALL software. The first, which is an important consideration for the course as a whole, is to be sure that the subject matter is linked to the needs and/or interests of the learners so that they are motivated to use it. The second consideration is to determine to what extent the content of the software is consistent with that of the rest of the course. If too much of the content is already familiar, learners will be less motivated to attend to it; on the other hand, if the concepts are too challenging or too dependent on unfamiliar background information, they may become frustrated. Finally, the linguistic level must be appropriate for the learners' current degree of proficiency. If the vocabulary and structure are too simple, then very little language acquisition may occur, even though the content may be mastered. Similarly, if there are too many unknown words or unfamiliar constructions, then the input may not be comprehensible enough to allow for much acquisition, and again, frustration may result.

Learner strategies that are most valuable to introduce and practice in a content-centered course, in addition to those mentioned above for a

notional-functional syllabus, are those that involve more efficient language production and comprehension. CALL software that promotes the use of reading-comprehension strategies is particularly appropriate, as is software that focuses on strategies for organizing compositions, assuming the overall curriculum and course requirements include a writing component.

In summary, the focus of this section has been on the criteria for evaluating software vis à vis its relative consistency with respect to the course or program syllabus and its orientation. The three most important criteria seem to be the following:

1. the approach manifested by the software is compatible with that of the syllabus type
2. the level and sequencing of the linguistic content is appropriate for the course as determined by the syllabus
3. the subject-matter content is appropriate for the goals of the course or program as determined by the syllabus and the presumed knowledge base of the learners

The degree to which the software meets these criteria will aid the evaluator in determining whether the CALL materials can be integrated effectively into the syllabus as basic or supplementary material.

A Supplementary Evaluation Form for Language Teaching Approaches

This final section of this chapter presents a three-part supplementary evaluation form for CALL software (Figure 2) that takes into account the characteristics of the three types of approaches and the learner-strategy considerations presented above, and describes a procedure for using it. Users adopting this form for in-house evaluation are encouraged to adapt and simplify it, selecting those categories considered most relevant and ignoring the rest. As the evaluation criteria represent to some extent the author's interpretation of each approach, it is also possible that an evaluator will want to expand or revise the form on the basis of a different interpretation of these central principles.

It is also important to realize that a teacher using this form may not find his or her teaching approach described or encapsulated within any single category. The direct method, for example, as described by Larsen-Freeman (28), includes characteristics of behaviorist approaches (with respect to pointing out and correcting errors as they occur) and acquisition approaches (with respect to providing comprehensible input and promoting learner–learner interaction). Thus, teachers who do not embrace a particular method exclusively are even more likely to find their assumptions about language learning included under more than one

category—and this is in itself instructive, for it demonstrates a fundamental eclecticism that many subscribe to. For the same reasons, using this supplementary form to evaluate CALL software is likely to demonstrate clearly that the approach underlying the software itself reflects characteristics of more than one category. Thus, using the form will reveal (1) the approach in which a given piece of software may find its greatest correspondence and most efficient use, and (2) a pattern of response across all categories that describe the approach subscribed to by the teacher.

It is unusual to find CALL software that rates highly in *all* the areas within a given category. Thus, the evaluation form will also provide information about less-than-ideal but nevertheless usable correlations between content of extant software and its intended purpose. Finally, the Supplementary Evaluation Form focuses *only* on the language-teaching aspects of the software, and it should therefore be used in conjunction with other forms or checklists such as those described in the first section of the chapter. Part I of the form holds principles that describe the three categories of approach—behaviorist, explicit, acquisition; Part II encompasses descriptors common to learner strategies; Part III is a short questionnaire for other pedagogical considerations.

I. *APPROACH CHECKLIST*

PRINCIPLE	DEGREE?	HOW WELL DONE?	COMMENTS
Behaviorist Approaches			
1. Presents vocabulary and structure appropriate to the learner's level	0 1 2 3 CJ	1 2 3 4 5 CJ	
2. Maintains the learner's attention on the task	0 1 2 3 CJ	1 2 3 4 5 CJ	
3. Will not accept errors as correct answers	0 1 2 3 CJ	1 2 3 4 5 CJ	
4. Requires the learner to input correct answer before proceeding	0 1 2 3 CJ	1 2 3 4 5 CJ	
5. Provides the learner with positive feedback for correct answers	0 1 2 3 CJ	1 2 3 4 5 CJ	
6. Provides sufficient material for mastery and overlearning to occur	0 1 2 3 CJ	1 2 3 4 5 CJ	
7. Subsequently reinforces patterns and vocabulary presented in earlier lessons	0 1 2 3 CJ	1 2 3 4 5 CJ	
8. Presents grammar patterns inductively without attempting to teach formulations of rules	0 1 2 3 CJ	1 2 3 4 5 CJ	

Explicit Learning Approaches

1.	Introduces or reviews grammar rules and word meanings in an understandable, learnable, and reasonably accurate form	0 1 2 3 CJ	1 2 3 4 5 CJ
2.	Provides effective practice so that (a) novel target-language input can be readily understood and (b) the learner's understanding of the rules studied leads to the production of grammatically correct spoken or written target-language forms	0 1 2 3 CJ	1 2 3 4 5 CJ
3.	Gives meaningful rather than mechanical practice	0 1 2 3 CJ	1 2 3 4 5 CJ
4.	Gives practice contextualized in a coherent discourse larger than a sentence	0 1 2 3 CJ	1 2 3 4 5 CJ
5.	Provides hints of various types to help lead students to correct answers	0 1 2 3 CJ	1 2 3 4 5 CJ
6.	Accepts alternative correct answers	0 1 2 3 CJ	1 2 3 4 5 CJ
7.	Provides explanations for why correct answers are correct	0 1 2 3 CJ	1 2 3 4 5 CJ
8.	Anticipates incorrect answers and provides explanations of them	0 1 2 3 CJ	1 2 3 4 5 CJ
9.	Maintains the learner's interest throughout the exercise	0 1 2 3 CJ	1 2 3 4 5 CJ
10.	Allows an appropriate degree of student control	0 1 2 3 CJ	1 2 3 4 5 CJ
	CJ = cannot judge	0 = not at all 3 = to a great extent	1 = poorly 5 = excellently

PRINCIPLE	DEGREE?	HOW WELL DONE?	COMMENTS

Acquisition Approaches

1.	Provides meaningful communicative interaction between the student and the computer	0 1 2 3 CJ	1 2 3 4 5 CJ
2.	Provides comprehensible input at a level just beyond that currently acquired by the learner	0 1 2 3 CJ	1 2 3 4 5 CJ

3.	Promotes a positive self-image in the learner	0 1 2 3 CJ	1 2 3 4 5 CJ
4.	Motivates the learner to use it	0 1 2 3 CJ	1 2 3 4 5 CJ
5.	Motivates the learner to acquire the language	0 1 2 3 CJ	1 2 3 4 5 CJ
6.	Provides a challenge but does not produce frustration or anxiety	0 1 2 3 CJ	1 2 3 4 5 CJ
7.	Does not include overt error correction	0 1 2 3 CJ	1 2 3 4 5 CJ
8.	Allows the learner the opportunity to produce comprehensible output	0 1 2 3 CJ	1 2 3 4 5 CJ
9.	Acts effectively as a catalyst to promote learner–learner interaction in the target language	0 1 2 3 CJ	1 2 3 4 5 CJ

II. *LEARNER STRATEGY CHECKLIST*

1.	Introduces the learner to strategies that are useful and immediately usable	0 1 2 3 CJ	1 2 3 4 5 CJ
2.	Introduces the learner to strategies appropriate to the learner's level	0 1 2 3 CJ	1 2 3 4 5 CJ
3.	Explains the value of the strategies	0 1 2 3 CJ	1 2 3 4 5 CJ
4.	Provides meaningful practice in the use of the strategies	0 1 2 3 CJ	1 2 3 4 5 CJ
5.	Presents practice material in such a way that the task is more easily or successfully accomplished if the appropriate strategy or strategies are used	0 1 2 3 CJ	1 2 3 4 5 CJ
6.	Provides an appropriate variety of strategies (or of techniques for utilizing a given strategy) for a given type of task suited to a range of learning styles	0 1 2 3 CJ	1 2 3 4 5 CJ
7.	Provides feedback on which strategies might have worked best for given tasks after the student has attempted them	0 1 2 3 CJ	1 2 3 4 5 CJ

CJ = cannot judge

0 = not at all
3 = to a great extent

1 = poorly
5 = excellently

III. *OTHER PEDAGOGICAL CONSIDERATIONS*

1. Based on your own interpretation of the ratings, what approach or approaches does this courseware most clearly represent?
2. How well does the software fit into the syllabus for the class or program for which it is being evaluated?
   a. Is the approach manifested by the software compatible with that of the syllabus?
   b. Are the level and sequencing of linguistic content appropriate for the course as determined by the syllabus?
   c. Is the subject-matter content appropriate for the goals of the course or program as determined by the syllabus and for the presumed knowledge base of the learners?
3. Briefly describe the methods and techniques used; comment on how successfully they have been adapted to the computer medium.

Figure 2. Supplementary Evaluation Form for Language Teaching Considerations

In the first two sections of the form (Approach and Learner Strategy), a two-part, scaled-response format replaces the simple yes/no format characteristic of the CALICO checklist and many others. The first part (Column 1) is a purely quantitative measure, scaled from 0 to 3, and represents the degree to which a certain criterion is met, with 0 meaning "not at all" and 3 meaning "to a great extent." The second part (Column 2) is meant to be a qualitative measure, scaled from 1 to 5, and represents a judgment on the part of the evaluator on how *well* the criterion is met, with 1 meaning "poorly" and 5, "excellently." Under both, the abbreviation CJ stands for "cannot judge." The expanded response scale offers these advantages: First, the scales provide more information than a binary format. Second, the "how well done?" column provides a way for judging the effectiveness rather than the mere presence of an option. For example, it is quite possible to have hints and explanations in explicit learning software that are copious but not particularly helpful, and this form is designed to indicate that information explicitly.

Going through the evaluation form will reveal that in come cases the judgments in the second part are unnecessary. For example, if the reviewer circles "0" for an item under the "Degree" column, then there is no sense in rating how well it was done. Further, for certain items, e.g., whether the material "motivates the learner to use it" (Acquisition Approaches, item 4), the rating in the "Degree" column should correspond closely to the rating under "How Well Done" simply because of the type of information requested. Note also that because scales alone may be misleading, there is a short space for comments after each item.

The third section of the form, Other Pedagogical Considerations, provides for a more open-ended response in three areas. The first question simply asks the evaluator to make a judgment about what approach the courseware under review represents most clearly. Note that because the principles may differ in importance to a given approach or to a practitioner of that approach, simply averaging the numbers in sections I and II

is unlikely to offer a reliable numerical index that defines approach. Rather, the rater will have to determine the answer to this question through a careful inspection of the categories themselves and an assessment of the relative weight of each item. Question 2 in section III covers the relationship between the software and the syllabus as outlined in the previous section. Question 3 asks for a brief description of the methods and techniques the software uses and an evaluation of how well they have been adapted to the computer.

It should be noted that this supplementary form is designed for CALL software that includes language content and is inappropriate without significant changes for evaluating word-processing programs and the like. Finally, there are judgments requested that may be duplicated in other parts of a more general checklist. This is a minor annoyance, however, and perhaps not much of a price to pay for seeing the information specific to a language-teaching approach represented in a more readily interpretable fashion.

A Procedure for Using the Evaluation Form

As mentioned in the opening section of this chapter, evaluating software for second-language instruction is a challenging task. The paragraphs that follow offer a procedure to aid an evaluator in completing that task successfully; its goal is to lead the evaluator to an informed decision about adopting a given piece of software.

Before beginning the evaluation process, it is necessary to acquire a general evaluation form like that in Figure 1, to which some version of Figure 2 above should be appended. The base form should include at least questions about the skill area, intended level, and hardware requirements, as well as whether the software requires or allows for student collaboration (see Wyatt, Chapter 4 in this volume), etc. As suggested earlier, the evaluator may also wish to adapt the Supplementary Evaluation Form in Figure 2 so that it includes only those areas deemed relevant for the class or program for which the software is being considered. For instance, if the software is to be used in a class taught strictly within a given approach, then the evaluator may not feel it is necessary to rate the fit of the software with other approaches and simply eliminate those categories. When the instructional focus encompasses a wider variety of approaches, the evaluator will need to use the entire checklist.

The actual evaluation process involves five steps, moving from a cursory level to a very detailed and critical review. As the goal is not necessarily to complete the form but to decide on whether to adopt a given piece of software, the evaluation procedure should normally stop at any point where the evaluator becomes convinced that the software is *not* appropriate for his or her class or program.

The first step is to examine the available documentation. The evaluator looks for general information such as the following:

1. language skill area and approximate level
2. form and content of the language material
3. types of student response required (multiple choice, completion, etc.)
4. types of feedback to the student available from the program
5. procedures for monitoring student progress and consequent adaptation of the lesson sequencing
6. record-keeping options available to the teacher

The next step is to skim through selected parts of the disk for additional information in the areas just mentioned and to get a sense of screen layout, clarity of instructions, and ease of use. A good strategy is to look at parts of at least three lessons taken from the beginning, middle, and end of the lesson sequence (if one exists). In multiple-disk packages this may mean looking at one lesson on each of three disks.

The third step is to take an in-depth look at one or more lessons (or parts of lessons if they are long). If more than one exercise type is available, then a sampling should be taken of each. At this stage, all the available user options should be explored. Hints and help options should be tried, and both correct and incorrect answers should be attempted with the evaluator noting the computer's responses and judging their appropriateness for the presumed users. It is at this time that many of the questions on the Supplementary Evaluation Form can be answered with a fair degree of reliability, and if the evaluator's time is an important consideration, a decision to adopt or reject the software can often be made. The information gathered from this and the previous step is also useful for determining whether the students will be able to use the software in its present form or will need additional documentation or instruction.

The final two steps are taken when additional information is either needed or desired: The fourth step is to move through the entire program from start to finish, essentially following the same procedure as in step three. The fifth step, which may be accomplished without going through the fourth, is to field-test the software on one or more students in the target audience. Any software that rates highly after these final two steps can be given a sound recommendation.

Conclusion

This chapter has stressed the importance of considering teaching approach in the evaluation of CALL software and has presented a Supplementary Evaluation Form to that end as well as a procedure for implementing it. The supplementary form provides teachers with access to the kind of information that allows for a more confident decision

whether a given piece of software is consonant with their assumptions about language learning, with the needs of students in their foreign language classes, and with the scope and sequence of their syllabus. While the focus of this chapter has been on the individual teacher's evaluation of CALL materials, there are two additional areas in which the concepts discussed above also find relevance: software reviews and software design.

If the evaluation parameters suggested in Figure 2 are valuable to individuals evaluating software for their own use, they are unquestionably important to reviewers. Due to time factors, the complexity of the evaluation process, and the difficulty in acquiring software for that process, published reviews play a vital role in the initial stage of deciding whether to consider given software for adoption or purchase. In some cases, for expedience, published reviews may even comprise the *only* information outside of the publisher's description that a buyer has access to prior to purchasing a software package. Thus, evaluators who publish software reviews need to address the sorts of questions found in the Supplementary Evaluation Form whether they use the actual form or not. In addition, journals and newsletters that publish reviews of CALL software could encourage the use of a checklist similar to that in Figure 2, or at base develop some guidelines for reviewers as a part of their editorial policy to ensure that *some* information concerning language-teaching approach is made explicit in the review proper.

The implications of focusing on language-teaching approach as the organizing principle in software design are also potentially quite significant. There are a number of specific recommendations that can be made to software authors and publishers, all of which stem from taking the principles listed in the Supplementary Evaluation Form as *design criteria*.

First, in the earliest stages of the design process, individual software developers and development teams should describe the language-teaching approach to be targeted; that is, determine the overall lesson structure and the role that graphics, sound, screen layout, etc., will play. By definition, this means that the author or one or more members of the designing team should have a solid background in contemporary second-language acquisition theory and research as well as a fair amount of teaching experience with the target audience. Too often, it seems, software has been authored on the basis of general CAI design principles alone or on insufficient language-teaching experience, with the result that a very limited and haphazard set of language-teaching principles is applied.

A second implication of widespread adoption of evaluation forms that include the type of information presented in Figure 2, especially in published reviews, is the potential to have a significant impact on commercial software packages. As was noted at the outset, current checklists and guidelines do not address specifically the kinds of evaluation parameters found in the Supplementary Evaluation Form. This fact suggests that potential buyers do not consider the areas described therein to be of much

importance, whereas checklists that include approach and strategy-specific parameters contain the message that these parameters *are* important. This could have two possible results. First, the publishers themselves may develop design criteria that include approach considerations, both for language software that they create directly and CALL software solicited from outside professionals. Second, once it is clear that evaluators and users consider this type of information important, publishers may become more consistent about including it in promotional brochures and in the documentation accompanying the software.

This chapter has described a rationale and a procedure to distinguish CALL software from other forms of CAI by something besides its subject-matter content. It is clear that many CALL practitioners, reviewers, and authors need to pay more attention to second-language acquisition and to teaching methodology in order to *apply* the insights from those fields appropriately in CALL software design, evaluation, and use. CALL is already on the cutting edge of technology: it can also be on the cutting edge of methodology. In order to achieve this, it is necessary to start with the concept that second-language learning is essentially *different* from other types of learning and that, consequently, the criteria for judging the pedagogical soundness of instructional courseware for mathematics, chemistry, or history are not sufficient for judging software for second-language instruction.

References, Language Teaching Approaches, the Evaluation of CALL Software, and Design Implications

1. Anthony, Edward M. "Approach, Method, and Technique." *English Language Teaching* 17 (1963):63–67.
2. Ariew, Robert. "Computer-Assisted Foreign Language Materials: Advantages and Limitations." *CALICO JOURNAL* 2, 1 (1984):43–47.
3. Asher, James J. "The Total Physical Response Approach to Second Language Learning." *The Modern Language Journal* 53 (1969):3–17.
4. Baker, Robert L. "Foreign Language Software: The State of the Art, or Pick a Card, Any (Flash) Card." *CALICO Journal* 2, 2 (1984):6–10,27.
5. Baltra, Armando, "An EFL Classroom in a Mystery House." *TESOL Newsletter* 18, 6 (1984):15.
6. Bialystok, Ellen. "A Theoretical Model of Second Language Learning." *Language Learning* 28, 1 (1978):69–83.
7. _____. "On the Relationship between Knowing and Using Forms." *Applied Linguistics* 3, 3 (1982):181–206.
8. Brooks, Nelson, *Language and Language Learning.* Orlando, FL: Harcourt Brace Jovanovich, 1980.
9. Brown, H. Douglas. *Principles of Language Learning and Teaching.* Englewood Cliffs, NJ: Prentice-Hall, 1980.
10. Brumfit, Christopher. *Communicative Methodology in Language Teaching: The Roles of Fluency and Accuracy.* New York: Cambridge University Press, 1984.
11. Carrell, Patricia L. "Three Components of Background Knowledge in Reading Comprehension." *Language Learning* 33, 2 (1983):183–207.

12. ———, and Joan C. Eisterhold. "Schema Theory and ESL Reading Pedagogy." *TESOL Quarterly* 17, 4 (1983):553–73.
13. Chomsky, Noam. *Topics in the Theory of Generative Grammar.* The Hague: Mouton, 1969.
14. Coady, James. "A Psycholinguistic Model of the ESL Reader," pp.5–12 in Ronald Mackay, Bruce Barkman, and R. R. Jordan, eds., *Reading in a Second Language.* Rowley, MA: Newbury House, 1979.
15. Curtain, Helen A. "Integrating Content and Language Instruction." *ERIC/CLL News Bulletin* 9, 2 (1986):1,10–11.
16. Dalgish, Gerard M. *Microcomputers and Teaching English as a Second Language: Issues and Some CUNY Applications.* New York: Instructional Resource Center, City University of New York, 1984.
17. Decoo, Wilfred. "An Application of Didactic Criteria to Courseware Evaluation." *CALICO Journal* 2, 2 (1984):42–46.
18. Dulay, Heidi, Marina Burt, and Stephen Krashen. *Language Two.* New York: Oxford University Press, 1982.
19. Ellis, Rod. "A Variable Competence Model of Second Language Acquisition." *International Review of Applied Linguistics and Language Teaching* 23, 1 (1985):47–59.
20. Grellet, Françoise. *Developing Reading Skills: A Practical Guide to Reading Comprehension Exercises.* New York: Cambridge University Press, 1981.
21. Higgins, John. "Can Computers Teach?" *CALICO Journal* 1, 2 (1983):4–6.
22. ———, and Tim Johns. *Computers in Language Learning.* Reading, MA: Addison-Wesley, 1984.
23. Hope, Geoffrey R., Heimy F. Taylor, and James P. Pusack. *Using Computers in Teaching Foreign Languages.* Language and Education: Theory and Practice, no.57. Orlando, FL: Harcourt Brace Jovanovich, 1984.
24. Hubbard, Philip. "Alternative Outlining Techniques for ESL Composition." *CATESOL Occasional Papers* 10 (1984):59–68.
25. ———, James Coady, John Graney, Kouider Mokhtari, and Jeff Magoto. "Report on a Pilot Study of the Relationship of High Frequency Vocabulary Knowledge and Reading Proficiency in ESL Readers." *Ohio University Working Papers in Linguistics and Language Teaching* 8 (1986):48–57.
26. Krashen, Stephen D. *Second Language Acquisition and Second Language Learning.* New York: Pergamon Press, 1981.
27. ———, and Tracy D. Terrell. *The Natural Approach: Language Acquisition in the Classroom.* San Francisco: Alemany Press, 1983.
28. Larsen-Freeman, Diane. *Techniques and Principles in Language Teaching.* New York: Oxford University Press, 1986.
29. Long, Michael H. "Does Second Language Instruction Make a Difference? A Review of Research." *TESOL Quarterly* 17, 3 (1983):359–82.
30. Madsen, Harold S., and J. Donald Bowen. *Adaptation in Language Teaching.* Rowley, MA: Newbury House, 1978.
31. Marler, Jerilyn, ed. *Evaluator's Guide for Microcomputer-Based Instructional Packages.* Eugene, OR: International Council for Computers in Education, 1982.
32. McLaughlin, Barry, Tammi Rossman, and Beverly McLeod. "Second Language Learning: An Information Processing Perspective." *Language Learning* 33, 2 (1983):135–58.
33. Mohan, Bernard A. "Language and Content Learning: Finding a Common Ground." *ERIC/CLL News Bulletin* 9, 2 (1986):1,8–9.
34. Newmark, Leonard. "How Not to Interfere with Language Learning." *International Journal of American Linguistics* 32, 1–2 (1966):77–83.

35. _____, and David A. Reibel. "Necessity and Sufficiency in Language Learning." *International Review of Applied Linguistics in Language Teaching* 6, 2 (1968):145–64.
36. Oller, John W. "Story Writing Principles and ESL Teaching." *TESOL Quarterly* 17, 1 (1983):39–53.
37. Oxford-Carpenter, Rebecca. "Second Language Learning Strategies: What the Research Has to Say." *ERIC/CLL Bulletin* 9, 1 (1986):1–4.
38. Rubin, Joan, and Irene Thompson. *How to Be a More Successful Language Learner.* Boston: Heinle & Heinle, 1982.
39. Scarcella, Robin C. "How Writers Orient Their Readers in Expository Essays: A Comparative Study of Native and Non-native English Writers." *TESOL Quarterly* 18, 4 (1984):671–88.
40. Spack, Ruth. "Invention Strategies and the ESL College Composition Student." *TESOL Quarterly* 18, 4 (1984):649–70.
41. Stevens, Vance. "Implications of Research and Theory Concerning the Influence of Choice and Control on the Effectiveness of CALL." *CALICO Journal* 2, 1 (1984):28–33.
42. Strei, Gerry. "Format for the Evaluation of Courseware Used in Computer-Assisted Language Instruction (CALI)." *CALICO Journal* 1, 2 (1983):43–46.
43. Strevens, Peter. *Teaching English as an International Language: From Practice to Principle.* Oxford, England: Pergamon Press, 1980.
44. Swain, Merrill. "Future Directions in Second Language Research," pp. 15-28 in Carol A. Henning, ed. *Proceedings of the Los Angeles Second Language Research Forum.* Los Angeles: University of California at Los Angeles, 1977.
45. Terrell, Tracy D. "A Natural Approach to Second Language Acquisition and Learning." *The Modern Language Journal* 61 (1977):325–36.
46. Underwood, John A. *Linguistics, Computers, and the Language Teacher: A Communicative Approach.* Rowley, MA: Newbury House, 1984.
47. Wenden, Anita L. "Learner Strategies." *TESOL Newsletter* 19, 5 (1985):1,4–7.
48. Wilkins, David A. *Notional Syllabuses.* New York: Oxford University Press, 1976.
49. Winitz, Harris, ed. *The Comprehension Approach to Foreign Language Instruction.* Rowley, MA: Newbury House, 1981.