**Martha C. PENNINGTON** holds a PhD in Linguistics from the University of Pennsylvania and is Reader in English at the City Polytechnic of Hong Kong. She serves on the editorial board of the *TESOL Journal* and has published in the *TESOL Quarterly*, *System*, *Computer Assisted Language Teaching* and *Language Culture and Curriculum*, and has edited collections on computer-assisted language learning and language programme evaluation.

**Vance STEVENS** holds an MA in English as a Second Language from the University of Hawaii at Manoa and is the Instructional Developer for CALL at Sultan Qaboos University in Oman. He has served on the Editorial Advisory Board of the *TESOL Quarterly* and is a contributing editor of *CAELL Journal* as well as Editor of the *MS-DOS Users' Group Newsletter* and Software Reviews Editor for the *CALL-IS Newsletter*. He has published in the *TESOL Quarterly*, *System*, *English for Special Purposes Journal*, and *CALICO Journal* and has jointly authored two CALL programmes with S. Millmore: SUPER CLOZE and TEXT TANGLERS (Research Design Associates).

**MULTILINGUAL MATTERS 75**
Series Editor: Derrick Sharp

# Computers in Applied Linguistics: An International Perspective

Edited by
Martha C. Pennington and
Vance Stevens

**MULTILINGUAL MATTERS LTD**
Clevedon • Philadelphia • Adelaide

**Library of Congress Cataloging in Publication Data**

Computers in Applied Linguistics: An International Perspective/Edited by Martha C. Pennington and Vance Stevens.
p. cm. (Multilingual Matters: 75)
Includes bibliographical references and index.
1. Applied linguistics - Data processing. I. Pennington, Martha Carswell. II. Stevens, Vance. III. Series: Multilingual Matters (Series): 75.
P129.0885 1991
418'.00285 dc20

**British Library Cataloguing in Publication Data**

A CIP catalogue record for this book is available from the British Library.

ISBN 1-85359-120-3 (Hbk)
ISBN 1-85359-119-X (Pbk)

**Multilingual Matters Ltd**

*UK:* Bank House, 8a Hill Road, Clevedon, Avon BS21 7HH, England.
*USA:* 1900 Frost Road, Suite 101, Bristol, PA 19007, USA.
*Australia:* PO Box 6025, 83 Gilles Street, Adelaide, SA 5000, Australia.



Typeset by Editorial Enterprises, Torquay.
Printed and bound in Great Britain by Billing & Sons Ltd.

# Contents

# Preface

This book grew out of an idea for a symposium entitled, 'Computers in Applied Linguistics: The Decade of the 1980s and Beyond', that we organised for the World Congress of the International Association of Applied Linguistics (AILA) at the University of Sydney, Sydney, Australia, in August 1987. Many of the papers represent second or third generation versions of papers read at the symposium or in other sessions at the Congress. Still others were solicited later, in order to round out the contents of the volume and to ensure representation of certain topics and areas of interest.

The development and editing of the papers was a major undertaking, requiring approximately two and a half years to complete. Besides a number of technical matters relating to the content of individual papers, it has been a huge task to edit so many papers at such great distances. Not only were the two editors situated approximately half way around the world from each other, but the contributors were spread out across Australia, Europe, and the United States. Looking back, we probably could not have conceived of a situation less conducive to good communication.

Though the process was slow and sometimes frustrating for us and the contributors, we did manage to communicate — via air mail, telephone, electronic mail, and personal travel. Each of the original papers has been extensively reworked based on two or more rounds of feedback from each of us. In their present versions, we feel that each paper represents a significant contribution to the burgeoning field of computers in applied linguistics, and we are pleased to have been able to support the development of this timely collection.

We wish to express our thanks to M. A. K. Halliday, whose encouragement to publish the papers from the AILA symposium led us to develop the present volume, and to Manfred Pienemann, who suggested Multilingual Matters as a potential publisher for the material. We also wish to express our thanks to all of our contributors for their interest in the project and willingness to work with us in bringing the volume to completion.

Martha C. Pennington
*Hong Kong*

Vance Stevens
*Sultanate of Oman*

# 2 Humanism and CALL: A Coming of Age

VANCE STEVENS
*Sultan Qaboos University, Sultanate of Oman*

## Introduction

This article attempts to put into perspective the coming-of-age of computer-assisted language learning (CALL), various aspects of which are documented in other parts of the volume. It is argued here that, in shaking off the influence of the early behaviourists, CALL is becoming more 'user-friendly', or humanistic. Whatever its precise manifestation, humanism in CALL means that courseware lends itself these days more to what students want it to be than what a particular program designer may have originally intended it to be. This watershed development has not only brought CALL more in line with current thinking about language teaching methodology, but also heralds the emergence of CALL as a versatile tool, as an aid to learning, and as an informant on language rather than a preceptor, task-master, or programmed instructor.

Before embarking on this topic, a few points should be made clear. First of all, it is not suggested here that CALL, or any learning for that matter, is not to some degree behaviouristic. Almost every human interaction — teaching in particular — is an attempt to modify behaviour. However, behaviourist psychologists have been restrictive in the transactions allowed in their models of how behaviour is modified, and it is the courseware evolving from such models that current CALL developers tend to reject.

A second point of clarification is that, in so far as no one theory succeeds in describing a mechanism as complex as how people learn — let alone how they learn languages, which themselves defy accurate description — this article does not intend to dogmatically espouse any particular theory of learning. It shows, rather, how recent developments in CALL exemplify humanist paradigms. Proponents of various learning theories (e.g. Skinner, 1974) generally

acknowledge the limitations of their theories while arguing that the available evidence supports their view of how people learn. The view put forward here is that since we lack a means of either accurately describing or controlling what it is that we want people to learn, one major advantage of CALL is its flexibility in providing students with tools that utilise their demonstrated abilities to figure things out and so to learn according to their motivation and interest.

In its role as tool in language learning, CALL is not assumed to be the sole means of delivery of a course of language instruction. Therefore, certain objections to learner-directed modes of study (e.g. that they are inefficient because they rely on induction and leave the learner to wander aimlessly in a jungle of disordered linguistic data) could, if desired, be compensated for in other parts of a course. In this way, CALL can be integrated into the curriculum as a medium that allows students to experiment with concepts taught elsewhere. Conversely, CALL is also something that a student can employ on an *ad hoc* or self-access basis without its being part of a set course of study.

## Changing Attitudes Toward the Way People Learn

It is widely recognised that mid-way through the present century, a convergence of behaviourist psychology and of structural linguistics led to a dominance of the audiolingual method in language instruction, and to interest in **programmed instruction** (PI) as a means of delivering instruction in general. Drill-and-practice CALL is in some respects reminiscent of the former, while PI lent itself particularly to the design of computer-assisted instructional programs (see Ahmad *et al.*, 1985, for a survey of this era of CALL development).

Eventually, as Brown puts it (1980: 242), 'language teachers were discovering that the ALM actually was not working! People were not learning the communicative functions of language.' A similar impression of programmed instruction is conveyed by Rivers (1981: 119), who notes that its tendency to 'preoccupation with the teaching of innumerable details about the language ... can distract attention from the real "terminal behavior", that is, the whole of language as it operates in an act of communication.' The problem with early CALL, as with the teaching practices it emulated, was that while both might have succeeded in teaching the surface forms of the language, neither promoted carry-over to spontaneously communicative contexts.

It was obvious to practitioners that either the model on which these methods were based or the interpretation of the these models in the creation of instructional algorithms was flawed. It may be unfair, as is often done, to lay the blame for these failures on 'behaviourism' as a whole; Carroll (1966: 104), in



fact, attributes to the audiolingual habit theory 'a vague resemblance to an early version of a Thorndikean association theory'. Nevertheless, general dissatisfaction with lockstep teaching encouraged an interest in more humanistic approaches to learners and their individual strategies for learning.

Humanist approaches to learning are articulated in the psychology of Carl Rogers. Rogers (1961: 35) postulates an:

> urge which is evident in all organic and human life — to expand, extend, become autonomous, develop, mature — the tendency to express and activate all the capacities of the organism, to the extent that such activation enhances the organism or the self ... it is my belief that it exists in every individual, and awaits only the proper conditions to be released and expressed.

Hence the humanistic educator's interest in environments conducive to learning which favour this quest for self-actualisation. Such an approach is epitomised for computer-based learning in Papert (1980).

According to Graham (1986: 56), Rogerian therapy provides a threat-free environment 'in which the individual learns to be free. As such it is an educational process, one that Rogers believes can be as effective within the classroom as in the clinic' (Rogers, 1969) and which depends on the relationship between facilitator and client having three significant qualities. The first is what Rogers (1961: 33) calls 'a transparency on my part' and what Brown (1980: 77) interprets, in the area of language learning, as a focus:

> away from 'teaching' and toward 'learning'. The goal of education is the facilitation of change and learning. Learning how to learn is more important than being 'taught' something from the superior vantage point of a teacher who unilaterally decides what shall be taught.

This description anticipates the second significant quality of the therapist or teacher, which is 'unconditional positive regard for the client', while the third is empathetic understanding or genuine listening — a continuing desire to understand the feelings and personal meanings which the person is experiencing' (Graham, 1986: 55).

Rivers (1981: 89) characterises the influence of humanist psychology on language learning as follows:

> In practice, the humanistic approach has resulted in the inclusion in language-learning materials of vocabulary and activities for expressing one's feelings, for sharing one's values and viewpoints with others, and for developing a better understanding of others' feelings and needs. A language class is a particularly suitable environment for meeting affective needs,

> because much of the activity can take the form of role playing, simulation games, and small-group discussions. The expressive arts ... require the student to seek the most appropriate forms in the new language to express nuances and meaning.
>
> With this re-emphasis on individual worth and difference, language teachers became conscious of the fact that individual students prefer different modalities of learning ... they also learn at different rates and employ quite different strategies for understanding and retaining the material to be learned. With this new understanding, teachers were no longer satisfied with a monolithic 'what is good for one is good for all' approach.

Complaints concerning the inadequacy of CALL software have surfaced frequently in the literature, but are generally directed against the kind of software based on instructional algorithms rooted in PI and the pattern practice techniques associated with audiolingualism (Hubbard, this volume, chapter 3, cites several such complaints). But the situation is changing; we are now seeing improvements in CALL courseware in accordance with more recent shifts in approaches to the way people learn languages (Stevens, 1989a).

The present article attempts to show how these improvements have followed from humanistic approaches to language learning, as characterised above. First of all, it will be shown how present CALL courseware emphasises learning rather than teaching. Recent CALL development attempts to exploit an urge for self-actualization in learners through use of non-threatening learner-centred settings in which discovery learning, problem-solving, and tool-based activities figure prominently. Furthermore, in current CALL courseware, the role of teacher is diminished with respect to that of the student, in providing control over modalities for learning and in attempting to compensate for individual differences. Second, the article suggests that current CALL courseware displays 'positive regard' for students in so far as programmers have gone to great lengths in their consideration of the convenience of those using it. Finally, the article raises the question of communication both with and around computers — in particular, regarding activities in which students can 'express nuances and meaning' in the target language.

## Focus on Learning as Opposed to Teaching

Humanism in language learning seeks ways to empower individuals to direct their own learning, rather than ceding control over learning to an authoritative entity, as in behaviourist models. Higgins (1983; 1988) expresses this notion in terms of his **pedagogue/magister** dichotomy. In his view, the magister is the instructor directing students unilaterally, while the pedagogue is a slave following a step behind, always ready with a clue or answer when asked. The pedagogue facilitates learning but does not control it. Higgins points out that magisters have their place in learning, but when 'magisterial thinking' predominates, learners may be discouraged from extending their learning through productive experiment.



Whereas it often serves the magister's purposes to customise the teaching environment, the pedagogue operates in the environment at large. Accordingly, and despite a dearth of corroborating evidence, many language teachers have found intuitively appealing Krashen's suggestion (1982) that grammatical competence will follow naturally from interesting, relevant, comprehensible, and unsequenced language input. More recently, parallel distributed processing (PDP) has emerged as a model of cognition which seeks to explain how humans learn without resort to explicitly expressed rules. Developers of PDP models have succeeded in simulating acquisition of past tense verb forms (Rumelhart & McClelland, 1986) by programming a computer to develop 'connection strengths which allow a network of simple units to act as *though* [emphasis in original] it knew rules' (McClelland *et al.*, 1986: 32). Such a model suggests a mechanism by which acquisition could conceivably occur solely through learner interaction with authentic language environments. The above paradigms lend theoretical support to use of materials for language learning which replicate an authentic target language environment.

The computer is coming to be regarded as a medium with significant potential for work with authentic materials. It is possible to store large databases containing natural language on computers and to provide students with means of accessing these more thoroughly and efficiently than is possible with other media. From a humanistic standpoint, interaction with the database is non-threatening and is prompted by learner interest, curiosity, and need in fulfilment of the urge to 'expand, extend, become autonomous, develop, mature'.

We look now at three genres of courseware promoting student autonomy in learning. Such software provides environments for (1) exploratory interaction, (2) problem solution, and (3) use of software tools to develop productivity skills contributing to greater maturity in language learning. Some evidence that students have favourable attitudes toward learning English using a battery of CALL software comprising these three genres has been presented by Stevens (1988b; 1989c).

## Exploratory Interaction

One school of CALL development, pioneered for language learning by Higgins and Johns (1984), sees the computer as a repository of information,

only parts of which are revealed at any given moment. Thus, students are led to supply the missing pieces according to their knowledge or intuition about how the target language operates. From the feedback received, students test their intuitions, strengthening or weakening them in a continual process. In so doing, students make discoveries about the target language. It is assumed that because they have to work things out according to logic and pattern, they are more likely to internalise and retain the fruits of their discoveries than with other media.

In diametric opposition to exploratory software is software following the programmed instruction approach. The PI approach utilises carefully prescribed steps incorporating a gradual build-up of knowledge, with checks on learning at each step, possibly including branching according to how well the student is doing. (Where one part of the instructional sequence must be mastered before the student proceeds to the next, this is more specifically referred to as **mastery learning** (see Jamison *et al.*, 1974, for an overview of research into PI and mastery learning).) With PI-based CALL, what is learned is largely controlled by the programmer. Exploratory software, on the other hand, provides tools enabling students to browse and manipulate a database. Access to information could be in almost any order, and its presentation is under the control of the student.

Both approaches have their advantages and disadvantages. The first approach is behaviourist-based and will result in language learning only if the programmer has an accurate model of all relevant aspects of the learning situation and can emulate this on the computer. For example, a model of the student would have to accommodate individual differences such as maturity level, proficiency in the target language, and preference for inductive or deductive approaches (addressed in greater detail below). The model of the material to be taught might be relatively simple, as in the case of verb forms in various languages, or complex, as in the case of how these same verbs might be used in discourse. The compelling thing about this approach is that it would work if — but only if — we understood and could control the relevant aspects of the learning (and linguistic) situation. The possibility of close control explains the popularity of such an approach at a time when behaviourists were fairly confident that steps in learning could be reduced to discrete elements, and when linguists felt that they were on the verge of developing productive grammatical models of living languages. However, those who take a more holistic approach regarding the interaction of the many complex and little understood cognitive and affective factors involved in language learning and who see little hope of devising truly accurate models of either languages or learners can never be reconciled to the PI approach as a way of developing competence in a language beyond the most rudimentary levels of proficiency.



It has been pointed out that one advantage to the exploratory approach is that students can be provided powerful means of systematically accessing data, and so can, in theory, learn by observing and manipulating more comprehensive and authentic databases of materials than they could using any other medium. While PI might actually limit student access to linguistic data, in so far as it restricts them to a prescribed program of learning, the exploratory approach broadens the field considerably and encourages student independence and curiosity. Another advantage to the latter approach is that, in purposely allowing flexibility, the materials developer does not require in advance full information on the targeted users or their learning situation.

Machine-readable authentic text is often available in the workplace in the form of materials that ESL instructors, or teachers in other disciplines, have created using word processors. Often, such texts are created for purposes other than CALL work – for example, course materials which students are studying concurrently with their courses at a language center, or even everyday correspondence. Students are thus able to work interactively with texts that are authentic and often of immediate relevance to them (see Stevens, 1988a, for a description of the use of such programs in a language learning setting).

Johns (1989) also notes the advantages of using relevant and authentic texts with language learners. His remarks are made with regard to concordancing, which is another example of exploratory language learning software. A concordance program will find every instance of the occurrence of a string in a given text and display the context surrounding each instance of that string. Sources for extensive corpora of such text are discussed by Sampson (this volume, Chapter 9). Long the domain of linguists and literature specialists, concordances are only recently finding their way into the second language classroom. Stevens (1989b) describes how concordance output can be directed toward creating relevant and authentic vocabulary exercises, including hands-on exercises whose purpose is to encourage ESL learners to gain insights into salient features of the target language by running their own concordances on samples of authentic discourse.

One of the most compelling applications of the exploratory principle is hypermedia (described more fully below) — for example, hypertext, which allows students to access facilitative information by opening 'windows' on individual words or chunks of text. The windows may themselves allow elaboration on text, so that the amount of information available can proceed to some depth. Two hindrances to development of this approach are the labour needed to prepare the elaborations to text and the memory needed to store these elaborations if help is to be global. The latter problem could be attenuated through use of CD-ROM, in which case storage memory is sufficient for dictionaries and similar resources to be made available for random access. The first problem then

becomes one of developing algorithms to enlist the available resources in making possible elaboration on demand (see Lian, this volume, Chapter 4 for insights on such algorithms). The result is text which students can read (or videos they can explore, or a combination of these and other media) with the help of instant-access dictionaries, encyclopeadias, maps and atlases, diagrams, translations into various languages, or whatever on-line assistance has been built into the program. It is easy to imagine how placing a cursor over a troublesome lexical item and having the computer provide an animation, illustration, or information pertaining to that item would be an improvement over students' tediously consulting off-the-shelf resource materials when reading or writing, a process so cumbersome as to be underutilised by students and others.

Success with the exploratory approach requires that students be sufficiently motivated to search the database and that they do so in a systematic manner. Therefore, in practice, the success of exploratory CALL depends largely on the extent to which students are guided and motivated to work efficiently, as well as on the power and flexibility of the exploratory tool and on the scope and authenticity of the database (see Esling, this volume, Chapter 11 for a report on one such implementation).

## Problem-Solving

The problem-solving format cuts across almost all aspects of CALL; accordingly, there are puzzle and problem-solving elements in many of the examples of software cited under other topics in this article. Still, this format is itself an important characteristic of humanistic software.

One type of software for exploring databases is that which permutes text, creating reconstruction puzzles which students then resolve. For example, sentences are put out of order, and students restore them; or sentences are encrypted and students decode them; or cloze passages are created, and students replace the missing words (e.g. Stevens & Millmore, 1987).

Another useful problem-solving genre is the adventure game. Computerised adventure games are based on algorithms that set up a 'maze' of possible outcomes which are accessed according to choices made by the player in pursuit of some goal; successful negotiation of the maze is the puzzle in need of solution. Cheung and Harrison (this volume, Chapter 8) discuss the motivating value of the adventure game format, which they refer to as 'move-based simulation'.

Surprisingly, few adventure games have emerged specifically for CALL. One such game is *LONDON ADVENTURE* (Hamilton, 1986), in which a



traveller faces a day in London with a shopping list of last-minute gifts to buy and a plane to catch. Armed with a certain sum in foreign currency travellers cheques, the player must ask directions politely of passers-by in order to find an exchange bureau to get pounds. The player must then buy maps and guidebooks providing information on London transport and department stores in order to purchase the gift items before it is time to catch the plane home (the program sets a clock at the start of play). Other CALL adventure games include the interactive video programs discussed below, as well as *MYSTERY HOUSE*, adapted for ESL by Baltra (1984), and now in the public domain. The latter is one of many mystery and adventure games created originally for native speakers but capable of providing productive problem-solving contexts for language learning.

Another genre of software that has come to frequently employ a problem-solving format is tutorial software. Such software often operates in Presentation and Play modes; that is, there is one mode by which students receive instruction (sometimes by perusing a database) and another where they play games presenting challenges based on what has been learned. For example, there is a commercial software package which presents various facts on explorers of the New World during the Age of Discovery and then sets up problem-solving tasks which rely for their solution on associating the correct explorers with given facts (Neosoft, 1984). Other such commercially-available programs deal with subjects such as geography (*EUROPEAN NATIONS AND LOCATIONS*, 1985) and science (*FAMOUS SCIENTISTS*, 1985). In promoting reading skills in the target language (especially scanning and culling desired information from a text), and in providing settings for the discussion of facts and solutions to problems, such programs can provide fruitful contexts for language learning.

Increasingly, programs for teaching grammar, vocabulary, reading and writing incorporate games and puzzle elements; *ESSENTIAL IDIOMS* (Richardson & Wise, 1985), for example, teaches idioms in English via activities ranging from simple presentation of the material to a beat-the-clock mode. The latter activity is essentially a variation on fill-in-the-blanks, but its appeal to students is greatly enhanced by use of a game-board format, and by making the clock an integral part of the puzzle (see Stevens, 1987).

An interesting combination of adventure and tutorial formats, with value and appeal to ESL students, is found in *ROBOT ODYSSEY* (Wallace & Grimm, 1986). The game starts with a student alter ego getting out of bed and falling literally to the bottom of Robotropolis, to the sewer. The rest of the game is a quest to extract oneself step by step up through the six levels of Robotropolis and back to the light of day. In order to evade the strict but otherwise benign police robots and effect an escape, one must work through a series of tutorials to learn how to

enter and rewire the robots. The tutorials teaching the skills required are computer counterparts to Total Physical Response (Asher, 1966); that is, they instruct students to move this, pick up that, put this there, etc. The student reads these instructions and learns by actually doing as instructed. Once a tutorial is completed, the student has the specialised skills necessary for escaping another level of the Robotropolis maze. Thus, the game proceeds with the student reading instructions, performing operations which at every step confirm comprehension of the instructions, and eventually using the newly acquired knowledge to resolve the current predicament toward the overall goal of escape from Robotropolis (Stevens, 1988c).

Other language learning puzzle games available for personal computers include interactive versions of popular board-format games such as *Scrabble* and *Trivial Pursuit*. Finally, the text manipulation programs mentioned earlier all rely on a human interest in puzzles as motivators for what might otherwise be relatively mundane text-based exercises. It is important to note that the effects of computer-based enhancements on such exercises would be impossible to achieve in any other medium.

Presenting language learning tasks in problem-solving format has its drawbacks, however. For one thing, puzzles do not appeal to all types of language learners; some may benefit from inductive (i.e. problem-solving, exploratory) modes of learning, even as others would prefer more directed and deductive modes (as discussed more fully below). Also, if students do not understand the underlying benefits to them of solving a particular puzzle, they may look upon the activity as a frivolous waste of time, or they may work the puzzles only as such and not approach the tasks in ways conducive to language learning. Care must be taken to ensure that puzzle activities are constructed so as to promote effective language learning strategies rather than allowing students to resort to puzzle-solving strategies that may actually be counterproductive for language learning.

There is little research on this aspect of CALL; however, Windeatt (1986) notes several ways that cloze exercises as they are typically implemented on computer may counter optimal reading strategies. For example, students working cloze exercises on a computer treat text locally rather than globally, as they rarely scroll past one screen (when the cloze was presented on paper, they tended to read over the entire text). Moreover, they tend to pursue solutions one blank at a time rather than considering other blanks which might provide clues to the solution of the original blank (students working on paper moved quickly from blank to blank).

Further evidence that the strategies students employ for solving puzzles may not involve strategies for language learning can be gleaned from observing



students solving cryptograms. Cryptograms are puzzles in which each unique letter in a block of text has been changed to some other unique letter selected at random; for example, all the occurrences of *a* become *c*, all occurrences of *b* become *p*, and so on throughout the block of text. Interestingly, students solving cryptograms can complete a puzzle yet be oblivious to the message in the sentence elucidated. This is evident when the encrypted message carries instructions, and the students, on successfully decoding the message, fail to follow the instructions and profess later to have been unaware that any instructions were given. In other words, they seem to be capable of decoding an encrypted sentence without attending to its meaning. This may be isolated behaviour particular to certain students, and perhaps with certain texts, but it is worth being aware of possible limitations to the use of puzzles in effective CALL implementations.

In sum, problem-solving can serve as an adjunct to language learning and can be more entertaining than explicitly instructional modes. However, development of appropriate language puzzles may require more study and effort on the part of materials developers and teachers than one might at first expect if the activity is to achieve the desired effects.

## Using Software as a Flexible Tool for Productivity and Learning

One important reason for the increasingly common acceptance of computers as a viable medium for language learning is that computers have high face validity, in that they are perceived by both students and teachers as being important to learn to use. Phillips (1986: 4) points out that the computer is 'now part of mass consciousness and permeates social life at many different levels', and unlike the language lab is capable of 'being transplanted to the environment of the world outside'. In other words, computers have the unique advantage of being viable instructional tools, and at the same time of being devices that students want to use and to become more familiar with for reasons quite apart from language learning. Thus CALL presents a unique opportunity to provide learners with a rich environment of functional, communicative, and interactive materials in a given target language, and to do so by means of a delivery system that students are often predisposed to use.

Accordingly, a working knowledge of computer-based tools, the third kind of software promoting autonomous learning, is useful in its own right. Common examples of such software are word processors, desk top publishers, communications software, computer-based spreadsheets, database management programs, spelling checkers, syntax and style checkers, and programs dealing explicitly

with various stages in the writing process. Learners using such software practice with language, both in learning to use and in actually using the software, and also experience a gain in *productivity skills* which can in turn be put to use in further enhancement of linguistic abilities.

Productivity skills enable the learner to use tools which enhance control over organisation and manipulation of data, largely by reducing what Kemmis *et al* (1977) called *inauthentic labour*, i.e. the energy one must expend to accomplish the authentic labour which is the true purpose of undertaking the task. An example of inauthentic labour (in Higgins, 1988) is retrieving a dictionary from among other books on a shelf, perusing the key words at the top of its pages, and finally searching the page itself for a particular word. The availability of an on-line dictionary reduces to a few key-presses such ancillary steps and helps ensure that most labour done is authentic — in this instance, the labour involved in discerning the meaning of the word in question (see Witton, this volume, Chapter 13 for an example of a similar implementation).

After surveying a number of CALL developers in the United States, Johnson (1985) concludes that the computer could be much better used 'as a tool to accomplish functional tasks' than as a vehicle for 'traditional or even communicative CALL'; and that the study of language when using computers should be a 'by-product' of instruction focusing on tasks related to social and academic success in school (pp. III-5 and III-6). Thus, software tools for improving productivity in work can be taught as foci for functional language learning activities. The communicative aspects are enhanced because the students have real reasons for seeking information directed at solving immediate problems and are motivated to learn the language they need in order to use the software for accomplishing specific tasks. Furthermore, foreign students at American universities and secondary schools, who often initially have difficulty making friends with native-speakers, can find native-speaking classmates willing to help them use the computers in student terminal rooms. Thus the chance of making contacts vital to assimilation into a foreign culture is enhanced by the need to seek help in utilising computers as productivity tools.

Several textbooks have appeared which teach computer-based productivity skills as CALL. For example, the text by Abdulaziz *et al.* (1985) teaches computer programming to ESL students. Similarly, Barlow (1987) has produced a book teaching skills such as typing, word processing, use of writing aids, and spreadsheet and database manipulation to advanced level non-native speakers. It is not difficult to envisage how such skills might be taught in a content-based language course. The focus would be on the computer-based productivity skill, but the target language would be the medium of instruction and of the materials used; and the students would meaningfully practice that language by interacting



and communicating with other users of the skill in question (both novice and experienced) toward accomplishing tasks that utilise that skill.

## Providing Control Over Modalities for Learning

The overriding importance of allowing the user rather than the program to control the progress of a lesson or activity is still not universally accepted in program design, as is evidenced by software still on the market which locks students into a sequence of steps dictated by the program (e.g. Rosen *et al.*, 1985). However, allowing learners control over the various aspects of their learning is an important consideration in humanistic CALL. In their essay on 'clarifying educational environments', which describes conditions especially conducive to learning, Moore & Anderson (1969) argue the importance of providing learners with several perspectives on learning, among which they can shift at will. Their remarks prompted a study by Stevens (1984) examining the benefits of letting students choose sequence of CALL delivery, with results suggesting that learning is enhanced when choice and control are in the hands of students.

Humanistic CALL programs provide not only a multiplicity of choices, but also quick and easy means of making them. For example, students should be able to move easily within (or out of) the program — there should be a clear means of escape from the program, either at once or by backing out through menus, and of escape from tasks, perhaps by 'passing' on to the next one. Students should be able to choose to receive hints or even answers to save them the frustration of getting stuck.

Software developers are increasingly displaying all options clearly on the screen, and usually in some consistently designated area so that the user knows where to look for them. Often, selection of options is accomplished by moving the cursor until the option is highlighted, and then pressing the Return or Enter keys, or clicking. Often, there is a space on the screen for a description of the purpose of the option currently highlighted, and this information changes as the cursor is guided back and forth over the options. The trend for current software is not only to provide more options for users, but to provide means within the programs themselves to ensure that users know what these options are.

## Compensating for Individual Differences

It has long been acknowledged that individualisation of learning is one advantage of CALL over other media. Traditionally, individualisation has meant

that the learner can choose to work in privacy and silence, or that the software will branch according to how well the student is doing with the content of the program. More recently, it has been suggested that individualisation in CALL could achieve greater sophistication than this.

In research considering a range of cognitive styles, abilities, and aptitudes in students undergoing a course in computer-managed instruction, Federico (1982) found a need for 'adapting instruction to individual differences in students' cognitive attributes' (p.17). Chapelle & Jamieson (1986), in research on ESL students using *PLATO*, decry the approach taken in those lessons as being 'notoriously "insensitive" to individual learner differences' (p.41) and suggest that this insensitivity may be a reason why field independent students disliked CALL implemented on the *PLATO* system. If field dependence or independence is indeed a factor in a student's appreciation of a certain exercise format, and if the computer can be programmed to judge a student's field dependence or independence, then it follows that students could be presented lessons in alternative formats commensurate with their cognitive styles.

This field of inquiry is a potentially burgeoning one for CALL. As Ellis notes (1985: 116):

> the existing research does not conclusively show that [cognitive style] is a major factor where [linguistic] success is concerned. There has been no research into the effects of cognitive style on route of acquisition.

Jamieson & Chapelle (1987) attribute this inconclusiveness to limitations in the methods of data collection employed for such research (observation and self-reporting). Because their data was collected on-line and analysed by the computer, Jamieson & Chapelle were able to show how the cognitive variables field dependence/independence and reflection/impulsivity are related to three learning strategies, and how the latter are predictors for TOEFL scores.

Not only do data collected on computer have a high level of integrity relative to the other means mentioned, but such data are in turn amenable to analysis by the computer (the data in the Jamieson & Chapelle study, for example, were rendered by means of a computer program to a format usable by the SPSS statistical software package). The ability of computers to control lesson delivery, scrupulously collect data on its use, and then analyse the results also made possible the studies by Doughty (this volume, Chapter 7) and van Els *et al.* (1988). The improvements in data collection and analysis made possible by computer-based research suggest techniques not previously available to researchers for probing the effects of cognitive style on the 'route' of second language acquisition.



The idea of testing for learner differences and then adjusting program delivery accordingly has occasionally been applied in CALL. For example, generative CALL lessons have been produced which individualise grammar materials based on error analyses of the students' first languages (Dalgish, 1989). Also Dunkel (this volume, Chapter 12) addresses the concept of tailoring CALL lesson difficulty according to learner proficiency.

Practically speaking, although we have the theoretical capability of catering to a wide spectrum of individual differences, and though work is in progress to actualise what is currently being imagined in theory, this area of CALL development is still much in the future. Consider, for example, the potential of bringing artificial intelligence and expert system techniques to bear on individualisation, as suggested by Lian (this volume, Chapter 4). Given the 'insensitivity' to individual differences of so much courseware in the past, the potential for improvement through greater individualisation of courseware is great.

## Positive Regard for Students: Humanistic Interfaces

Current CALL courseware is programmed with 'positive regard' for students to the extent that it is considerate of their comfort and requirements when using it. This is evident in considering recent developments in interfaces to computer-based learning tools, whose power is increased in part through increased convenience of access made possible by improved interfaces.

Conceiving of interfaces is like delving into dreams, as this mental process can easily transcend the realm of reality. Evans, in *The Mighty Micro* (1979), imagined himself lying on his back in bed staring at a terminal fixed to his ceiling as he paged about the snippets of reading matter which he had just downloaded over phone lines from the central library database. When Evans wrote his book, the mouse had not been invented, but this would have allowed him, by moving his fingers over a device at his bedside table, to perform such tasks as opening windows, blocking text off, and clicking it in to save and read later, etc.

Hofstetter (1985) has pointed out how important it is that what people have to do to get the computer to function should not distract them from what is happening on the screen. The recumbent Evans, staring concentratedly at the effects of his prestidigital manipulations on the screen on his ceiling, was implicitly aware of this fact. Having to look up and down at a keyboard is distracting, whereas keeping eyes focused on a screen while using a touchscreen capability (as provided in the *PLATO* system), manipulating a mouse or a joystick, talking to the computer, or pointing at the screen is not. Peripheral devices are, however,

not commonly available on student computers although such devices on personal computers in homes and offices are increasingly common.

Designers of interfaces seek to render non-purposeful (i.e. inauthentic) effort negligible. An example of an interface that significantly reduces the labour in marking linguistic data is reported in Pienemann and Jansen (this volume, Chapter 10). Similarly, Esling (this volume, Chapter 11) notes how using computers 'makes it possible to access many pertinent items of data that would otherwise be extremely difficult, or even impossible, to extract from an interfering matrix of irrelevant material'. What would make such a task impossible is of course the amount of inauthentic labour involved.

The interfaces that Esling describes involve the attachment of audio devices to the computer. Audio interfaces allowing the computer to speak or even to recognise speech have potential for reducing distraction in student interaction with machines (encouragingly, Dunkel, this volume, Chapter 12, notes that audio interfaces are becoming cheaper and more readily available).

Hypermedia, in conjunction with a variety of peripheral media, has recently caught the attention and imagination of CALL developers. The original hypermedium was hypertext, which allows computer users to indicate a word on the screen (by moving the cursor to it) and to get more information on that word, perhaps a definition or an example (usually by opening a window on the additional information). This basically entails embedding 'buttons' anywhere on the screen and making things happen when the buttons are 'clicked'. In the possibilities for making things happen, recent implementations of hypermedia have gone well beyond text.

In Underwood's (1989: 8) view, hypermedia implementations 'give the student power over the medium: the power to explore a body of information without being constrained by the author's view of how it all fits together, the power to follow an idea as far as one's imagination, and the medium, will allow. ' In the implementation Underwood describes, the computer controls a videodisc player. The Main Menu is a picture of a room with objects in it, each of which is a HyperCard button. Clicking the preview button near the television set, for example, calls up a map of the story-line of the film on videodisc. 'The power of the map, however, is that it is alive: each of the icons representing the scenes is also a HyperCard button linked to a set of videodisc player commands' (Underwood, 1989: 16). Clicking any button causes a ten-second preview of the scene, after which the script of that scene in Spanish can be seen. Icons alongside the script allow students to play the whole scene or to see an English translation, and clicking on any line in the dialogue or its translation causes that line to be played from the videodisc. Other buttons present exercises, or allow the students to access the database of scripts. Thus, students can search for any word



in the script, or for every occurrence of any word, as in a concordance program (except that here, they can also play the accompanying video segment on demand).

Another provocative example of hypermedia is Ashworth & Stelovsky's (1989) *KANJI CITY*. Students must read signs, maps, and business cards in order to negotiate the fictional Kanji City, and must finance their journey by passing quizzes at the school or winning at a grammar slot machine. In simulating social encounters, *KANJI CITY* provides opportunities for learning social behaviour as well as for improving language skills. Following yet another trend in recent CALL development, the program constitutes an integrated set of shell programs which will work on their own or with data sets designed to teach languages other than Japanese.

It is also possible, using only the standard keyboard, for software designers to ensure focused attention to the computer screen. This is accomplished when user input is restricted to the cursor movement keys, space bar, and Enter or Return key, all of which can usually be struck without having to look down. Encouragingly, software making imaginative use of these keys in user interfaces is becoming commonplace.

For example, the *LONDON ADVENTURE* game mentioned earlier allows display of options by using the arrow keys to present these in the manner of a revolving drum (where each option over-writes the last one in the same place on the screen, as if the options were written on a drum which, when turned, displays the next or previous option). Similarly, *SUPER CLOZE* (Millmore & Stevens, 1990) lists options on a screen with an arrow that runs up and down the list in response to the press of any of the following keys: the arrow keys, the space bar, the Enter key, or a number key corresponding to the number next to the item in the list — whichever the student finds most convenient. While the student peruses the options, an explanation of the one highlighted appears below the list in the manner of a revolving drum. As in the case of *LONDON ADVENTURE*, selection of any option is made by pressing the Enter key; it should therefore not be necessary for students to look down from the screen to rummage through the options and make a selection. Still other commercial programs in the adventure game format allow arrow keys to control movement of a figure around a screen layout representing rooms, stairs, a river (choosing a direction at a fork), and the like (Klug & Relf, 1984; McKinley & Ragan, 1984; Stine *et al.*, 1984).

As further accommodation to the way people generally prefer to use software, manuals (though often provided) are frequently not required to learn how to work the most recent software. Such software either has an on-line tutorial to help the user learn how to use it, or, as is often the case, menu screens that guide

the user by suggesting options at obvious junctures. Recourse to manuals is further obviated by providing help on demand; and the help given at a particular moment might even refer specifically to the task or screen at hand, or even to the current cursor position. This is quite in keeping with the humanistic trend toward greater consideration for learners in making CALL software convenient to use.

## Communication in and Around CALL

A final characteristic of a humanistic learning environment is that it encourages free expression of feelings and opinions and creates an atmosphere conducive to subtle nuance in communication.

Students engaged in computer-based activities often form groups around the computer. This is in part because computers promote brainstorming in resolving the outcome of interactional sequences, and in part because exploratory interaction creates opportunities for using language to discuss with teachers and peers the nature of discoveries made in the course of completing computer-based tasks. In addition to communicating with others while using computers, student interaction with the computer itself can be to some degree 'communicative'. Accordingly, two forms of communication prompted by computers can be distinguished: (1) that between language learners and others working in the computer-based interactive environment, and (2) that between language learners and the computer itself.

## Communication Between Language Learners and Others in the Computer-using Environment

That computer-based activities can be an impetus for communication among students as well as between students and teachers has often been noted anecdotally (e. g. Dutra, 1985; Taylor, 1986) and is observable whenever two or more students engage in CALL at the same computer, or when students ask for help in using computers or in solving problems posed by the software. Rivers (1989) agrees that 'practical use has shown that [task-oriented games and simulations] provide for genuine communicative interaction when students work together in groups at the workstation, the challenges of the CALL activity stimulating them to lively discussions, disputes, and cooperative decision-making'.

In computer labs, students tend to form groups of two or three around a single computer, even when there are enough computers available for each student to use one individually. One reason for this may be that students using



computers do not feel that they are being watched or judged; perhaps as a result they do not feel that the work they do on the computer is their own private property. When they do not regard computer-based activities as tests, they become relaxed about pooling information and seeking help from friends.

Brainstorming occurs when many minds are focused together on solving a problem. Everyone uses slightly different approaches and strategies when solving problems, and when learners work together in unravelling problems set by the computer, the contributions of different members in the group, each approaching the problem in a slightly different way, can help the group overcome obstacles. For example, in working on an adventure game, single players, using their limited repertoire of problem-solving strategies, can bog down, just as a driver stuck in sand might dig a car more and more deeply into a rut by persisting in the same strategy for trying to solve the problem. In the case of the driver, a passer-by may come along and suggest letting some air out of the tyres, or help gather wood or stones to put under the wheels. In the same way, one member of a student group will often suggest a move that is obvious to that person, but not to the others. Sometimes, the group will try it, and it works. One benefit of a group configuration is that members can eventually learn from each other alternate approaches to solving problems. Especially relevant to the discussion here, work at the computer puts students in situations where they seek advice and information from one another.

When using exploratory software, if the students are a homogeneous language group, they tend to use their own language in these groups; if instead their native languages are different, they will likely communicate with each other in the target language. Mohan (this volume, Chapter 6) provides insights into the kind of interaction that takes place in such heterogeneous groups. Leaving aside the exact nature of the communication that occurs, it is obvious that computers at least promote social aggregation, a vital prerequisite to communication. Other computer-based settings — e. g. electronic mail and computer bulletin boards — provide opportunities for communication in written form in the target language. The fact that computers seem to be a catalyst for human-to-human communication is worth exploring, and exploiting, in language learning.

Word processing is possibly the most universally used and at the same time the most universally underrated genre of communications software available today, although a survey by Johnson (1987) found word processors to be prevalent in ESL settings. Suggestions abound for using word processors as devices for teaching grammar and rhetoric. Schcolnik (1987) suggests a number of manipulations to writing that can be accomplished by means of word processors, such as garbling text with global search-and-replace commands and then having students put the text right by reversing the process. Daiute (1983) cogently

examines how word processing facilitates writing, while Pennington and Brock (this volume, Chapter 5) provide evidence that word processing facilitates meaningful revision in a process approach to writing. Marcus (1983) notes how word processors can become catalysts for oral, in addition to written, communication. He describes an activity, for example, in which students exchange computer screens, so that one student can comment on the other's writing while the latter is composing it blind.

Daiute (1985) also suggests numerous applications of word processing to language learning — for example, having students produce yearbooks using word processors or desktop publishing tools. In an ESL course in which two separate heterogeneous groups of students were trained to use the same word processor, the present author observed the students to become more involved than usual in preparing (and more willing than usual in revising) writings for yearbooks in which they communicated their feelings about their language learning experience. Additionally, they had to communicate regarding editing and layout with each other and with the company that would print their yearbook. The students worked without access to desktop publishing software, but such software in conjunction with a publishing activity would add a further communicative dimension to the activity, as students negotiated with experts and with each other in learning and then manipulating the software.

Another configuration that promotes communication in a target language allows students to be in direct contact with each other (or with teachers, or even with unknown correspondents) through networks. For example, John Southworth (in *ad hoc* demonstrations at the University High School in Honolulu, Hawaii) has used *PLATO* to put his students in touch with peers across the ocean. Similarly, internationally played simulations sponsored by the International Simulation and Games Association are reported in Crookall *et al.* (1988) and Crookall and Oxford (1988).

The above examples show various ways that communication between humans can be promoted by CALL. We now turn to communication within CALL itself, that is, to having a student in simulated communication with the computer.

## Communication between language learners and the computer itself

Before embarking on this discussion, we should consider carefully the notion of communication with computers. To say that people communicate with each other (whether or not around a computer) is to say that they desire to make themselves known to another communicator and to induce a response.



In communication, people exhibit verbal and written behaviours, the purpose of which is to invoke a response. The response desired could be phatic, informative, or motor. But more than that, it will likely be appropriate and empathic; that is, humans (at least when being polite) generally respond to other humans by doing more or less what is asked of them and by trying to appear willing to please (Grice, 1975). Above all, humans generally consider the feelings of others, and if they are unable to respond in the way expected, will frequently explain their reasons for not being able to do so.

Computers are able to communicate half-way — that is, they can prompt phatic, informative, and motor responses from students and can respond in kind to prompts; but they cannot do so with empathy. Being logical and mechanical, they could be programmed to perfectly emulate human verbal behaviour only if there existed an accurate model that could fully predict all such behaviour. Since none exists, nor is in the foreseeable future likely to exist, computers, even when programmed to be on their best and most civilised behaviour, inevitably respond to humans in ways that appear inappropriate and abrupt. Programmers are developing algorithms in the realm of artificial intelligence (AI) which are based on a variety of approximate models of human verbal behaviour, and these achieve varying degrees of success in attempting to communicate with humans. For example, highly communicative parsers are components of the expert-system CALL environments envisaged by Lian (this volume, Chapter 4).

Although there has been recent progress in AI-based CALL (e. g. Bailin & Thomson, 1988; Cook, 1988), the level of communication that users of present-day software can expect is limited. *ELIZA*, for instance, figures heavily in Underwood's (1984) book on communicative CALL in a chapter entitled 'The computer as a communicative environment'. *ELIZA* is a program designed to emulate an empathic listener. Not unexpectedly, however, *ELIZA* has no comprehension whatsoever of what the user tries to tell it, but works by locating key words in user input and fishing questions and statements from its database in reaction to these. Thus it is not really empathic, nor a listener in any sense at all.

Nevertheless, students have been known to communicate in good faith with an *ELIZA*-like program called *LUCY*. Stevens (1986) describes an incident in which a student tried to teach *LUCY* the name of the capital of Japan. The student began by asking the computer several times in succession what he thought was a simple question — 'What is the capital of Japan?' — to which the computer replied with a series of responses to incomprehensible questions; e. g. 'Why do you ask?', 'Does that question interest you?', etc. The student, a little annoyed, decided to inform the computer, by entering the information from the keyboard, that the capital of Japan was Tokyo, and then re-ask the same

question. When the computer reacted with the same intransigence as before, the student called in the teacher, who explained what the program was doing. Two healthy outcomes stemmed from this incident: (1) the student learned that computers were mechanical, not communicative, as he had previously thought, and (2) the teacher realised that *LUCY* was eliciting real communication, if only for a few minutes, from at least one student.

Despite limitations in how students are presently able to communicate with computers, the medium is still capable of greater communicative interchange than is possible with any other educational medium, save another person. This is especially true when computers are used in conjunction with other media. Because of its ability to sustain interest and give the learner command over the medium, interactive video seems particularly able to achieve a high degree of simulated communication. Saint-Leon (1988) has suggested that using authentic materials on *videodisc* might be the next best thing to learning a language in a country where the target language is spoken. With the high degree of control learners have with computer-interfaced videodisc, interactive video could, for the period of time that it is used, be an even more productive learning environment than residency in a foreign country (see, for example, Underwood, 1989, and the discussion below).

*MONTEVIDISCO* (Gale, 1983; Schneider & Bennion, 1984) is remarkable among projects simulating communication using videodisc in its liberal use of absurdity and humour. *MONTEVIDISCO* was filmed in Mexico with native speakers of Spanish operating in their natural environment. Students constantly confront people who speak to them in Spanish, after which they are given multiple choice responses. Students can play the program 'straight', or they can indulge in the bizarre — e. g. they can accept a potion from the local pharmacist and see the glass tilt before their eyes and the pharmacy ceiling swim out of focus, blending into that of the hospital where they seemingly 'come to'. This option leads to a confrontation with a nurse, whom students can obey and so stay in the hospital, or elude and then find themselves on the street. Walking the streets in *MONTEVIDISCO*, it is possible to turn a corner and come face to face with a cigar-chomping Mexican motorcycle policeman for one of the most memorable confrontations in the program (the policeman is not an actor, but the genuine article). This may lead to a trip to jail, where students can attempt to bribe the local authorities. This results in even more trouble — or fun, as the case may be. But whatever the results, they come about as an effect of students simulating communication with characters in the program.

An even more ambitious effort involving interactive video is the Athena Project being carried out at the Massachusetts Institute of Technology (Kramsch *et al.*, 1985). The Athena Project aims to bring high technology — i.e. advanced



parsing techniques and speech recognition, in addition to interactive video — to bear on authentic and language-rich learning materials. The umbrella project of Athena has spawned sub-projects, one of which is Furstenberg's (1987) *FRENCH VIDEODISC*. In this program, students assume the role of a resident of Paris who faces two challenges: (1) a deadline on a job and (2) the need to find a new place to live after falling out with a girlfriend. In accomplishing the latter task, students have access to on-line resources such as maps of Paris, newspaper advertisements, etc. One can also arrange a meeting with a real estate agent and receive a tour of a set of flats in various price ranges. Some of the flats are well beyond the hero's means, and the dialogue with the agent can become humorously sarcastic. For realism, the walking-tour was filmed in an actual Parisian neighbourhood and allows minute exploration; most if not all of the scenes were shot on location in Paris. Here again, students influence outcomes by communicating in the role of the main character.

Adventure games do not have to include sophisticated parsing in order to be communicative. *MONTEVIDISCO* communicates with the learner in natural and wholly contextualised Spanish, but the learner communicates with *MONTEVIDISCO* in single keypresses denoting multiple choice responses. Yet the impression of communication is such that the learner rarely feels constrained. *LONDON ADVENTURE* operates along the same lines, but without a video component. The student interacts with the computer by using arrow keys to view options comprising things to do and say; impolite and otherwise inappropriate utterances result in people turning away and going on about their business, while appropriately worded requests elicit positive responses and desired results. Although students do not actually compose utterances or even words in interacting with the program, they still control what is being said, and they constantly deal with the program's communications to them.

Although computers are not yet natural communicators, when programmed well, they are good at communicating with students, and often allow students to communicate in satisfying ways with them. Video by itself is able to communicate to a viewer; interfaced with a computer, video allows perhaps the ultimate in modern-day emulation of authentic communication.

## Conclusion

Rivers (1989) expresses her impression of the present state of CALL in the following advice to developers:

> There is a real danger of a return to much drill and grammatical practice, with long explanations, which are relatively easy to program ... They may

> provide the student with a more solid base of knowledge of the language but little opportunity to consolidate creative control for the expression of personal meanings.... It is essential ... that computer-assisted language learning set its sights much higher than the widespread drill-and-practice tutorials and grammar teaching, which in their common form, can be recognized as the strong influence of uncreative programmed instruction.... If CALL is to realize its professional potential, we must keep emphasizing the need for innovative thinking in the production of a diversity of materials that promote creative use of language by the learner.

In making courseware more humanistic, software developers are gradually removing the grounds for recently voiced dissatisfaction with CALL. In avoiding the rigid behaviourist-based approaches of earlier years, CALL is moving toward granting learners greater control over their own learning. It does this by providing exploratory environments for language learning, presenting problems in need of resolution, and providing tools for further work and learning. Present-day CALL seeks to provide multiple paths to learning and to cater more actively to individual differences than is possible with other media.

The future promises greater interactivity with computers in language learning. Exciting possibilities exist for interface with audio and video cassette recorders, as well as with CD-ROM and laser videodisc, especially when applying to all of these the greater potential for interactivity using artificial intelligence. Even today, CALL software developers are developing facilitatory techniques that make software more powerful by being more convenient and easier to use. More to the point — and exactly the point at which the behaviourist-based algorithms have largely failed — humanistic CALL software is designed to promote communication, either with the computer itself or with other students using the CALL program.

The case has been made here that these developments fit comfortably into a humanist paradigm. Of greatest importance, the shift away from behaviourist-based software and towards humanistic applications has brought CALL more into line with modern principles of language learning and teaching. As a result, CALL can now be more widely perceived as a welcome enhancement to contemporary language learning curricula.

## Acknowledgements

I owe many thanks to Martha Pennington, whose meticulous feedback on the various drafts led to many improvements in this paper; also to Alan Beretta and Roger Griffiths, who constructively played Devil's Advocates.



## References

ABDULAZIZ, M. , W. SMALZER & H. ABDULAZIZ (1985) *The Computer Book: Programming and Language Skills for Students of ESL*. Englewood Cliffs, NJ: Prentice-Hall.

AHMAD, K. , G. CORBETT, M. ROGERS & R. SUSSEX (1985) *Computers, Language Learning and Language Teaching*. Cambridge: Cambridge University Press.

ASHER, J. (1966) The learning strategy of the total physical response: A review, *Modern Language Journal*, 50, 79–84.

ASHWORTH, D. & J. STELOVSKY (1989) *KANJI CITY*: An exploration of hypermedia applications for CALL, *CALICO Journal*, 6: 4, 27–39.

BAILIN, A. & P. THOMSON (1988) The use of natural language processing in computer-assisted language instruction, *Computers and the Humanities*, 22, 99–110.

BALTRA, A. (1984) An EFL classroom in a mystery house, *TESOL Newsletter*, 18: 6, 15.

BARLOW, M. (1987) *Writing with Computers*. La Jolla, CA: Athelstan.

BROWN, H. D. (1980) *Principles of Language Learning and Teaching*. Englewood Cliffs, NJ: Prentice-Hall.

CARROLL, J. (1966) The contributions of psychological theory and educational research to the teaching of foreign languages. In A. VALDMAN (ed.), *Trends in Language Teaching*. New York: McGraw-Hill.

CHAPELLE, C. & J. JAMIESON (1986) Computer-assisted language learning as a predictor of success in acquiring English as a second language, *TESOL Quarterly*, 20: 1, 27–46.

COOK, V. (1988) Designing a BASIC parser for CALL, *CALICO Journal*, 6: 1, 50–67.

CROOKALL, D. , A. CECCHINI, A. COOTE, D. SAUNDERS & A. DELLA PIANE (eds), (1988) *Simulation-gaming in Education and Training*. Oxford: Pergamon.

CROOKALL, D. A. & R. OXFORD (eds), (1988) *Language Learning Through Simulation*. Rowley, MA: Newbury House.

DAIUTE, C. (1983) The computer as stylus and audience, *College Composition and Communication*, 34, 134–145.

— (1985) *Writing and Computers*. Reading, MA: Addison-Wesley.

DALGISH, G. (1989) Error analysis and courseware design, *C. A. L. L. Digest*, 5: 5, 1–3.

DUTRA, I. (1985) Hypothesis-testing and problem-solving software for ESL students. Paper presented at the 19th Annual TESOL Conference, New York, April 8–14.

ELLIS, R. (1985) *Understanding Second Language Acquisition*. Oxford, UK: Oxford University Press.

*EUROPEAN NATIONS AND LOCATIONS* (1985) San Francisco, CA: DesignWare.

EVANS, C. (1979)*The Mighty Micro*. London: Victor Gollancz Ltd.

*FAMOUS SCIENTISTS* (1985) Dimondale, MD: Hartley Courseware, Inc.

FEDERICO, P. (1982) Individual differences in cognitive characteristics and computer-managed learning, *Journal of Computer-Based Instruction*, 9: 1, 10–18.

FURSTENBERG, G. (1987) A videodisc for teaching French: Work-in-progress report. Paper presented at the Fourth Annual CALICO Convention, Monterey, CA, April 6–10.

GALE, L. (1983) *MONTEVIDISCO:* An anecdotal history of an interactive videodisc, *CALICO Journal*, 1: 1, 42–46.

GRAHAM, H. (1986) *The Human Face of Psychology: Humanistic Psychology in its Historical Social and Cultural Context*. Milton Keynes, UK and Philadelphia, PA: Open University Press.

GRICE, M. (1975) Logic and conversation. In P. COLE and J. MORGAN (eds), *Syntax and Semantics*, Vol. 3, Speech Acts. New York: Academic Press.

HAMILTON, T. (1986) *LONDON ADVENTURE*. Cambridge, UK: The British Council in association with Cambridge University Press.
HIGGINS, J. (1983) Can computers teach?, *CALICO Journal*, 1: 2, 4–6.
— (1988) *Language, Learners and Computers: Human Intelligence and Artificial Unintelligence*. London and New York: Longman.
HIGGINS, J & T. JOHNS (1984) *Computers in Language Learning*. Reading, MA: Addison-Wesley.
HOFSTETTER, F. (1985) Perspectives on a decade of computer-based instruction, 1974–84, *Journal of Computer-Based Instruction*, 12: 1, 1–7.
JAMIESON, J. & C. CHAPELLE (1987) Working styles on computers as evidence of second language learning strategies, *Language Learning*, 37: 4, 523–544.
JAMISON, D., P. SUPPES & S. WELLS (1974) Alternative instructional media, *Review of Educational Research*, 44: 1, 1–67.
JOHNS, T. (1989) Whence and whither classroom concordancing? In T. BONGAERTS *et al.* (eds), *Computer Applications in Language Learning*. The Netherlands: Foris Publications.
JOHNSON, D. (1985) Using Computers to Promote the Development of English as a Second Language. A report to the Carnegie Corporation of New York.
JOHNSON, N. (1987) Current uses of computers in ESOL instruction in the U. S., *CALICO Journal*, 5: 2, 71–77.
KEMMIS, S. , R. ATKIN & E. WRIGHT (1977) *How do Students Learn?* Working papers on computer-assisted learning: UNCAL evaluation studies. Occasional Publications 5. Norwich, UK: University of East Anglia, Centre for Applied Research in Education.
KLUG, R. , & P. RELF (1984) *TALES OF ADVENTURE*. New York: Scholastic.
KRAMSCH, C. , D. MORGENSTERN & J. MURRAY (1985) An overview of the MIT Athena Language Learning Project, *CALICO Journal*, 2: 4, 31–34.
KRASHEN, S. (1982) *Principles and Practice in Second Language Acquisition*. New York: Pergamon Press.
MARCUS, S. (1983) Real-time gadgets with feedback: Special effects in computer-assisted instruction, *The Writing Instructor*, 2, 156–164.
MCCLELLAND, J. , D. RUMELHART & G. HINTON (1986) The appeal of parallel distributed processing. In D. RUMELHART, J. MCCLELLAND, & the PDP RESEARCH GROUP (eds), *Parallel Distributed Processing: Explorations in the Microstructures of Cognition*. Vol. I, Foundations. Cambridge, MA: MIT Press; pp. 3–44.
MCKINLEY, A. & A. RAGAN (1984) *TALES OF DISCOVERY*. New York: Scholastic.
MILLMORE, S. & V. STEVENS (1990) *SUPER CLOZE*. Shareware available from TESOL/CALL-IS MS-DOS User's Group.
MOORE, O. & A. ANDERSON (1969) Some principles for the design of clarifying educational environments. Reprinted in C. GREENBLAT & R. DUKE (eds), *Gaming-simulation: Rationale, Design, and Applications. A Text with Parallel Readings for Social Scientists, Educators, and Community Workers* (1975). New York: John Wiley and Sons.
NEOSOFT, INC. (1984) *THE SEA VOYAGERS*. Greenwich, CT: CBS Software.
PAPERT, S. (1980) *Mindstorms: Children, Computers, and Powerful Ideas*. New York: Basic Books.
PHILLIPS, M. (1986) CALL in its educational context. In G. LEECH & C. CANDLIN (eds), *Computers in English Language Teaching and Research*. London, New York: Longman.
RICHARDSON, W. & S. WISE (1985) *ESSENTIAL IDIOMS IN ENGLISH: CAI ADAPTATION*. New York: Regents/ALA.



RIVERS, W. (1981) *Teaching Foreign Language Skills*, second edition. Chicago, London: The University of Chicago Press.
— (1989) Interaction and communication in the language class in an age of technology. Plenary address at the Fortieth Meeting of the Georgetown Round Table on Languages and Linguistics.
ROGERS, C. (1961) *On Becoming a Person: A Therapist's View of Psychotherapy*. London: Constable.
— (1969) *Freedom to Learn*. Columbus, OH: Chas. E. Merrill Publishing Co.
ROSEN, M. , B. LOWERY & M. MERRILL (1985) *THE WRITING SERIES*, Disks 1 through 5. San Francisco, CA: EduWare.
RUMELHART, D. & J. MCCLELLAND (1986) On learning the past tenses of English verbs. In D. RUMELHART, J. MCCLELLAND, & the PDP RESEARCH GROUP (eds), *Parallel Distributed Processing: Explorations in the Microstructures of Cognition*. Vol. II, Psychological and biological models. Cambridge, MA: MIT Press.
SAINT-LEON, C. (1988) The case for authentic materials on videodisc, *CALICO Journal*, 6: 2, 27–40.
SCHCOLNIK, M. (1987) Re-word CAI: A method for practicing reading skills with the aid of a word processor. Paper presented at the Fourth Annual CALICO Convention, Monterey, California.
SCHNEIDER, E. & J. BENNION (1984) Veni, vidi, vici via videodisc: A simulator for instructional conversations. In D. WYATT (ed), *Computer-assisted Language Instruction*. Oxford, New York: Pergamon.
SKINNER, B. (1974) *About Behaviorism*. New York: Vintage Books edition, 1976.
STEVENS, V. (1984) Implications of research and theory concerning the influence of control on the effectiveness of CALL, *CALICO Journal*, 2: 1, 28–33, 48.
— (1986) Using *LUCY/ELIZA* as a means of facilitating communication in ESL, *TESOL Newsletter*, 20: 2, 13–14.
— (1987) Software review of CAI adaptation of Robert J. Dixson's *Essential Idioms in English, TESOL Quarterly*, 21: 3, 558–564.
— (1988a) Self-access language learning materials at Sultan Qaboos University, *The Journal of Educational Techniques and Technologies*, 21: 2/3, 2–4.
— (1988b) CALL in a self-access resource centre. Paper presented at the 22nd Annual TESOL Conference, Chicago.
— (1988c) Software review of *ROBOT ODYSSEY* 1, *CALL-IS Newsletter*, 5: 1, 8–10.
— (1989a) A direction for CALL: From behavioristic to humanistic courseware. In M. PENNINGTON (ed), *Teaching Languages with Computers: The State of the Art*. La Jolla, CA: Athelstan.
— (1989b) Classroom concordancing: Vocabulary materials derived from authentic, relevant text. *English for Special Purposes Journal*.
— (1989c) A study of student attitudes towards CALL in a self-access student resource centre. *System* 19, No. 2.
STEVENS V. & S. MILLMORE (1987) *TEXT TANGLERS*. Stony Brook, NY: Research Design Associates.
STINE, W. , M. STINE & R. KLUG (1984). *TALES OF MYSTERY*. New York: Scholastic.
TAYLOR, M. (1986) Software review of *PRESIDENTIAL CAMPAIGN*, *TESOL Newsletter*, 20: 4, 12.
UNDERWOOD, J. (1984) *Linguistics, Computers, and the Language Teacher: A Communicative Approach*. Rowley, MA: Newbury.
— (1989) HyperCard and interactive video, *CALICO Journal*, 6: 3, 7–20.
VAN ELS, T. , K. DE GRAAUW & M. STORTELDER (1988) Grammatical feedback in a

computer-based editing task. *Australian Review of Applied Linguistics*, 11, 176–188.

WALLACE, M. & L. GRIMM (1986) *ROBOT ODYSSEY* I. Menlo Park, CA: The Learning Company.

WINDEATT, S. (1986) Observing CALL in action. In G. LEECH & C. CANDLIN (eds), *Computers in English Language Teaching and Research*. London and New York: Longman.

# 3 A Methodological Framework for CALL Courseware Development

PHILIP HUBBARD
*Stanford University, USA*

## Introduction

By the mid 1980s, microcomputers had been around long enough for many people to have had enough contact with them to recognise their potential as a tool for language teachers and learners. In addition, professional organisations devoted specifically to exploring the role of computers in language teaching had been formed, such as the CALL Interest Section of International TESOL (Teachers of English to Speakers of Other Languages); MUESLI, the microcomputer users groups of IATEFL (International Association of Teachers of English as a Foreign Language); and CALICO (Computer Assisted Language Learning and Instruction Consortium). CALL appeared to be on the ascendancy. But this period also saw the publication of a number of books and articles criticising the questionable effectiveness of commercially available materials. Ariew (1984), Baker (1984), Dalgish (1984), Holmes (1984), Pennington (1986), and Underwood (1984), among others, all argued that the current state of CALL was far from what it should and could be, and that the dominance of drill-and-practice software, often based centrally on behaviourist learning principles, was coupling the technology of the 1980s with the methodology of the 1960s.

Since that time, there has been some development in the areas of simulations, games, and grammar and reading tutorials, but the pedagogical value of much of the current software remains questionable. What is often overlooked in the criticisms, however, is the fact that we have been struggling to master a new technology that as a field language teaching was — and is still — unprepared to deal with. It should not, in fact, be surprising that CALL is still searching to find its place in language teaching, and that the initial uses of computers in this domain would be fairly obvious ones. Given the relative