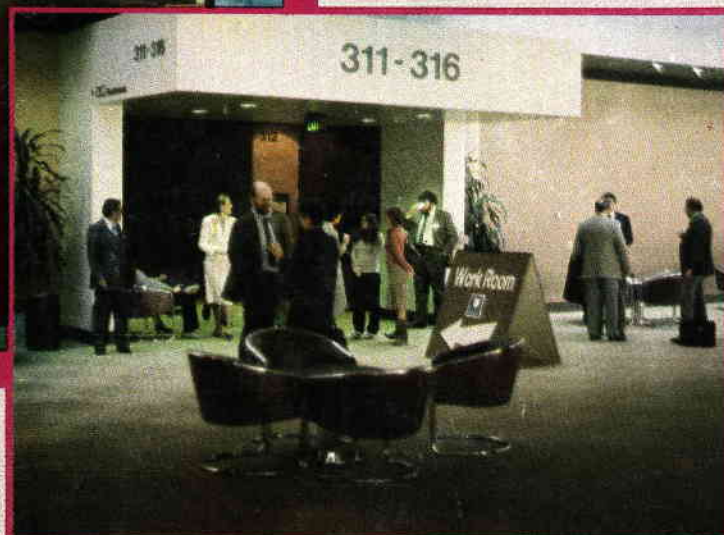
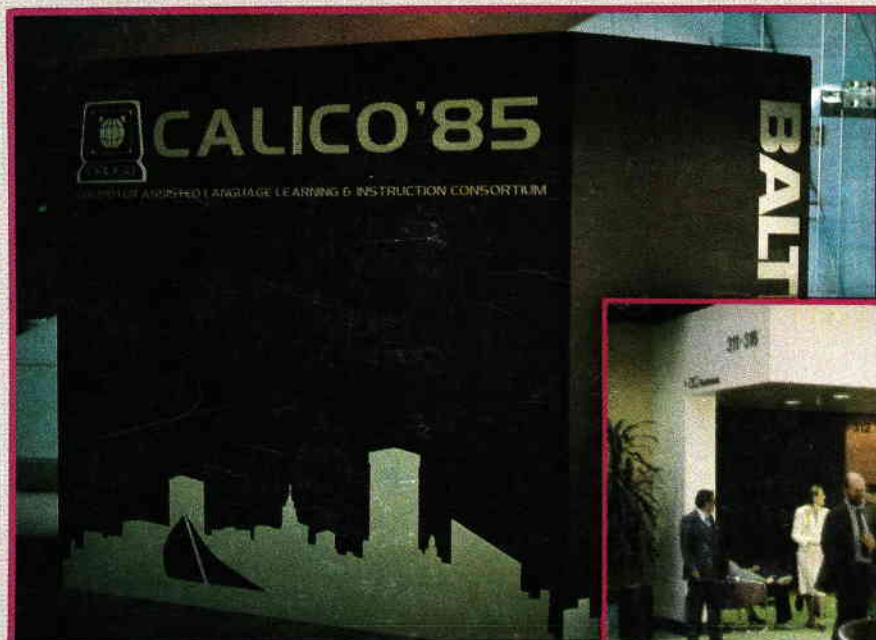


CALICO JOURNAL

CALICO *Symposium '85*



CALICO JOURNAL

COMPUTER ASSISTED LANGUAGE LEARNING & INSTRUCTION CONSORTIUM



VOLUME 2, NO. 3

MARCH 1985

Jungle or Paradise?	Minnie McNeal Kennny	3
Machine-Assisted Translation (MAT) SIG Report	Alan Melby	6
Courseware Development (CD) SIG Report	David Weible	8
Letter from the Executive Director: A Progress Report	Dr. Frank R. Otto	9
Baltimore City Schools Use Computers to Teach Writing	Charles R. Hancock	13
Special Thanks to Exhibitors		17
Computer-Assisted Instruction in Beginning College German: An Experiment	Herman U. Teichert	18
Artificial Intelligence (AI) SIG Report	Lori Levin	25
Book Review	Vance Stevens	27
Interactive Audio/Video (IAV) SIG Report	A. Allen Rowe	28
Computerized Adaptive Language Testing: Moving Beyond Computer-Assisted Testing	Jerry W. Larson Harold S. Madsen	32
Foreign Character Font (FCF) SIG Report	Kim L. Smith	37
CALICO Calendar		25
Courseware Reviews		38
Interactive Language Simulation Systems: Technology for a National Language Base	A. Allen Rowe	44
CALI/CALL Notes		48
Index to Advertisers		48

Reviewed by Vance Stevens

Vance Stevens directs the Institute of English Studies at Hawaii Preparatory Academy. He has done research into various aspects of CALL, and has been responsible for CALL implementations both at HPA and at the University of Petroleum and Minerals in Saudi Arabia.

Higgins, John, and Tim Johns. 1984.
Computers in Language Learning.
Reading, Mass.: Addison-Wesley
(Copublished with Collins ELT,
London). 192 pages.

Computers in Language Learning is one of the most important works to have emerged last year on how computers can and ought to be used in language learning. The book successfully addresses and combines these two interdependent elements; that is, (1) how computers ought to be used in language learning (the philosophy behind their use, and by implication, how they shouldn't be used), and (2) how computers can be used (what the programs will look like both to the user and to the programmer).

Despite the authors' avowal that their book was written mostly to the level of computer novice language teachers, the book can in fact be approached at two distinct levels of sophistication. The reader who is just beginning a study of computers in language learning will most likely be impressed with the variety of ways suggested here in which computers can facilitate the study and acquisition of language. Such a reader will enjoy a rich and descriptive exposure to the latest trends and techniques in CALL. More advanced readers, on the other hand, who may already be aware of many of these trends and techniques, will be able to appreciate the book in greater depth. In addition to rounding out their general knowledge of CALL, these readers will be able to plumb the program listings in the last half of the book for insights into novel strategies for specific CALL implementations.

The book is organized in such a way that the two levels of readers are not unduly burdened with information directed at readers in the other level.

Those already familiar with CALL might feel patronized by the half dozen pages at the beginning of Chapter 2 (sample headings: Screen versus printer, Screen size, The keyboard), but these readers will understand that the information here is necessary to a complete picture, and they might even appreciate being reminded of axioms of good CALL programming, such as The responsibility for making programs 'crash proof' and 'user friendly' is the programmer's. (p. 25) True novices will enjoy reading straight through to the last chapter, where program listings are elucidated. At this point, the novice is in for some rough going, and may prefer to skim for salient features, or to bail out and tackle this section after having gained more experience in the field. The programming instructor, on the other hand, will want to examine this section closely.

One problem with the last section, especially for American readers, is that the programs are written in Sinclair BASIC, a dialect that runs only on the Spectrum (TS2000), or with some changes, on the ZX81. Neither of these machines is widely used in America, and the fact that the programs will not run as-is on computers common in this country has been noted as a source of frustration in many reviews of this book published to date. Indeed, in the annotations accompanying the program listings, Higgins and Johns assume some familiarity with the structure and metalanguage of programming, rendering those parts of the discussion esoteric to the novice. Furthermore, machine specific oddities creep in, as when one finds at the end of a fascinating description of auto-grading a subroutine for saving the output to cassette, twice, in case one copy is corrupted. (p. 119).

Programming teachers, however, will find it more efficient to attempt a transposition than to reinvent these fancy wheels from scratch. Some of the dialectical differences are obvious; for example CLS to clear the screen may be accomplished by another command, for example HOME in Applesoft. As another example, the CODE command, which to a Sinclair registers the ASCII code of a string, can be done with similar commands in other varie-

ties of BASIC. Other differences are overtly noted; for example, Higgins and Johns point out that the TO command in Sinclair is a simpler way to use LEFT\$ in other dialects of BASIC.

Lack of transportability is an inconvenience which would exist for some users no matter what dialect of BASIC were chosen. The reader who focuses instead on the algorithms presented will come away more satisfied. For example, one algorithm is presented which, as a stand-alone program, allows students to explore environments for the determiners A and AN (pp. 72-3). As a subroutine in another program, the same algorithm checks syntax in text printed on the screen (pp. 122-3). Readers who attempt to understand the algorithms presented here are sure to pick up a trick or two which will enhance the effectiveness of their own CALL programming.

One of the unique and beneficial aspects of this book is the presentation of the theoretical framework in which Higgins and Johns see CALL as most effectively functioning. The authors seem most interested in exploring how the computer can further communicate, as opposed to linguistic competence in learners, and how it can facilitate acquisition, as opposed to learning (in the Kraschen sense). They discuss how the computer can both train students according to a pre-established regimen and how it can encourage students to explore language by breaking away from the more common paradigms of instruction. If they seem to be more interested in the latter possibilities, it is mainly because there are the more challenging, and often overlooked, ways computers can be used in modern language classes.

Higgins and Johns place computers at the new, and often unexplored, frontiers of language learning. They discuss how computers function well in even the more exhausted modes of instruction, but note that there is potential for abuse here. It is thus with barely veiled sarcasm that the authors warn: The computer is an obedient beast and will readily take on the role of drudge if required to. (p. 9) Higgins and Johns have convincingly shown here numerous ways by which that undesirable outcome can be avoided.