

Research Design Associates, Inc./Mind Builders

This Something About Us is written by the President of Research Design Associates, Inc./Mind Builders, (P.O. Box 848, Stony Brook, New York, 11790, 1-800-654-8715), Dr. Bob Leonard.

RDA/Mind Builders grew from an idea. Four years ago, I was an English teacher at a Long Island high school with an itch to own a computer to do word processing. As I began, my twenty years' experience in the classroom convinced me that, computer or no, many students would still have the same problem of coming up with what to say. So for them, the computer would be another "blank page" they'd have to fill. I started wondering about a program which would teach writers to generate, explore, and expand ideas before they began to write a draft.

I asked the high school computer coordinator if a computer could be programmed to take writers through the prewriting process. The computer coordinator, John Marchisotto, now my partner and RDA's Vice President said, "Why not? Let's try it." From that came *Proteus—the Idea Processor*, our first product which has won awards for three years.

I tell you this story because it really explains a cardinal precept in our company's philosophy. All RDA/Mind Builders software programs are and will be designed by teaching professionals to address real classroom needs. They also, where appropriate, have authoring systems so teachers can tailor the content to the specific concerns of their students and curricula. We feel it is an error to produce a program and then say, "Here it is, now find a use for it."

Another important belief we have is that the computer has the potential to be interactive, that it is far more than an "electronic textbook," so

all of our programs are interactive, and give feedback and avoid the "skill and drill" syndrome. Software should address and teach students to employ higher levels of thought, beyond rote learning, and those thinking skills should be transportable into the student's academic life.

Before a program goes out into the field, we test it with real students and modify it based on what we see and what they say. The final test is mine. It's a standing joke around here that we'll sell any program I can run with no difficulty. Classroom teachers don't have the time to spend hours poring over manuals. A program has to have a very rapid learning curve, with the manual being almost superfluous.

Finally, we believe that software should be affordable enough that every student will have the opportunity to use it, and that purchasers are entitled to updates if we improve the product. We initiated a Site License policy which allows unlimited duplication of our products at a cost that every school can afford and a lifetime guarantee that gives our customers the very latest version of the software free.

RDA/Mind Builders is still a small company. In fact, we're a "cottage industry." We still don't have a full-time secretary (I typed this) or a corporate headquarters, but that's O.K. We have faith that the quality of our product line is so great that we'll succeed. In the past year we've grown from a three product to a twenty-two product publisher. The authors we publish, like John and Muriel Higgins and Vance Stevens (See article by Vance Stevens, "Self-Access Language Learning Materials at Sultan Qaboos University," in this issue) are proven leaders in their fields. And best of all, the feedback we have received from the academic community assures us that we're publishing "the right stuff."

Text Tanglers: A Vance Stevens RDA/Mind Builders Software

Vital Statistics

Courseware Name: Text Tanglers
Application: English as a Second Language
Instructional Method: Decoding
Vendor: RDA/Mind Builders
Cost: \$39.95
Copy Policy: No copying allowed
Equipment: IBM or compatibles
Prerequisites: 2 disk drive or 1 drive & hard disk

General Description

Basically, *Text Tanglers* (TT) offers a set of activity driver programs into which English as a Second Language (ESL) teachers can insert their own texts for students to work with. The activities which TT enables are **Hangman**, **Cryptogram**, **Jumbled Sentences**, **Jumbled Paragraphs**, and **Cloze**; students are also allowed simply to read through any of the texts included in the text files. Twelve texts come with the program, and the teacher can easily create and add others as desired; any file so added can then be the basis for any of the activities.

The programming is, for the most part, competent with the notable exception of scoring the game-like activities. Although the student is provided with a lot of information about the activity and about his or her progress through it, the pedagogical approach is the overall problem with this programming. If the activities presented in *Text Tanglers* appeal to a teacher as worthwhile learning tasks, this package may be quite attractive. This reviewer suspects, however, that many teachers will not want their students to spend much time doing this kind of task.

The Activities

In **Hangman**, one word at a time from the text selected appears as the number of question marks representing the correct number of letters. The alphabet appears across the middle of the screen; when the student guesses a letter which is, in fact, represented in the word one or more times, it replaces the appropriate question marks. An incorrect answer brings forth a buzz and a piece of the scaffold. Letters guessed correctly appear below their place in the alphabet in green; incorrect guesses appear in red. The student thus can keep track of what has already been guessed. The student can ask to see the correct answer, after which the program advances to the next word, or the student can ask for a hint, after which **Hangman** suggests one of the correct letters. Status information includes the number of words available in the text, how many have already been attempted, and scoring information (to be discussed later).

In **Cryptogram**, one sentence from the selected text appears in an alphabetic code in which each letter has systematically been "encrypted" as another. Below the sentence an equal number of dots appears. Below that, the alphabet is lined up across the screen in a fashion similar to that found in **Hangman**. The student puts the cursor on any letter in the code sentence and makes a guess as to what it really is; if correct, the letter automatically appears in *all* its locations in the sentence, and the real letter appears beneath its counterpart in the alphabet line. If incorrect, the buzz sounds. The student can ask to see the solution (in which case the program advances to the next sentence), or the student can ask for a hint about the letter in the

code under which the cursor is located. Hints consist of the statement "X has been encrypted as Y," where X is the "real" letter and Y is the code equivalent. The reviewer found this confusing, since Y is what one is asking about. "A Y in the code stands for X" would be a more helpful, less confusing hint. If the user inadvertently asks for a hint on a letter which has already been provided, the program gives the hint which refers to the next unguessed letter in the sentence. Again, this can be quite confusing.

In **Jumbled Sentences**, the sentences from the selected text appear on the screen one at a time in the order in which they are found in the text. On the first screen, the sentence appears correctly with the directions "Please study this sentence." With the next key press, it appears in scrambled word order. Capital letters and punctuation remain attached to the words as they appear in the original. The student uses the arrow keys to move highlighting from one word to another, pressing ENTER to select a word. If the word selected is the next one in the original sentence, it disappears from the jumble and reappears below in order. If not, the buzz sounds but nothing happens. The Hint suggests the next word, and the solution is available.

In **Jumbled Paragraph**, the paragraphs appear in the order in which they are found in the text, one paragraph at a time. The individual sentences are presented as a list. The student uses the up and down arrow keys to select a sentence which is then highlighted, presses ENTER, uses the up and down arrow keys again to select a new location for it, and presses ENTER again to transpose to the new order. When the student thinks the sentences are all in order, he or she presses F2. Up to the point at which F2 is pressed, the program accepts all the student's moves as if they were correct; sentences can be rearranged for as long as the student wants to rearrange them. If F2 is pressed, however, and the order is incorrect, the feedback message is "You could be correct, but this is not the original sentence order." This message acknowledges that more than one order may sometimes be equally plausible. If the choice is correct, both sound and flashing messages "reinforce" it. If a Hint is requested, the program automatically highlights the first sentence in the list which is not in correct

position and suggest *that* sentence be moved down lower, but without indicating how much lower. This may cause confusion in the minds of some students because it may cause students to keep switching the order of the first two sentences and continually get the Hint to move each one lower down. The solution shows the paragraph in its original form. At this point, the student may choose between going back and doing the same paragraph again or the next one.

In **Cloze**, sentences from the selected text appear in order on the screen, one at a time, with every fifth word replaced by a blank with the number of dots corresponding to the number of letters in the missing word. The cursor moves automatically to the first dot of the blank, and when that word has been correctly entered, it jumps to the next blank. If the word typed in is not correct, the "wrong" buzz sounds when ENTER is pressed, and all letters disappear; the cursor goes back to the first dot. Blanks can be filled in out of order, but if the student forgets to press ENTER after typing in a word and manually moves the cursor to the next blank, the just-typed letters disappear. In this activity, the Hint gives the letter directly over the cursor; the cursor then moves to the next dot. Again, the user can ask to see the solution or the whole sentence.

General Design Features

Text Tanglers is a very colorful package. This reviewer found it to be almost too colorful even though the colors are used consistently throughout. Audience response to the colorfulness of *Text Tanglers* will depend on age and level of sophistication. The sound effects are generally unharmonious, strident, and unpleasant. Once an activity commences, the sound can be turned off, but rebooting returns to the default setting (sound on) and the cacophony which accompanies the introductory logo cannot be avoided.

The menu offers the user choices: first, the activity; then, the text on which the activity will be based. The user is always provided with a great deal of information: the activity and text chosen, the number of items in an activity, and the number of correct and incorrect responses

(see discussion of scoring below). A good range of options is provided: students can call for a help screen which spells out the directions for each activity, for a Hint, and for the correct solution, can turn off the sound, or can quit the activity. The latter returns the program to the main menu, and here, unfortunately, there is no way to quit to DOS, only the option to start over.

Programming

The package is competently programmed. The keys designated for offered choices seem to be always active; the unneeded keys are inoperative. This reviewer found no bugs in the program, but there are a few inconsistencies: the machine sound responding to a correct response is not always the same, and the message "Good-bye" appears sometimes when an activity is abandoned.

Scoring

The worst feature of *Text Tanglers* is the scoring and should, in this reviewer's opinion, be eliminated. Teachers should certainly not base any evaluation of a student's success or perseverance based on the reported score; they should tell their students to ignore the scoring. For example, in several of the activities, repeating a correctly guessed letter will increment the score infinitely; repeating an incorrect answer will increment the negative score. Except for **Hangman**, the activities are not particularly game-like, but rather exploratory. Even if the scoring mechanism in *Text Tanglers* were better, it would not add anything to the learning experience.

Inserting One's Own Texts

Text Tanglers allows teachers to create their own text files for students to use in any activity. Texts may be typed out using any word-processor

that can create pure ASCII files, i.e., files without any formatting characters. Not a programmer, this reviewer found text insertion relatively easy to do and add to the available list. The documentation is adequate but generally not well-written. Teachers without any computer experience, however, will probably need help until they are used to the routines.

Use for Foreign Languages

Unfortunately, this package is not designed for use with any language but English. Although the program does allow the use of foreign language texts in the jumbled sentences and paragraph activities, the problem is that the programs neither allow for typing in foreign words nor for accented characters. In principle, the package could be easily adapted for use with foreign languages whose diacritical marks can be created with regular ASCII code.

Pedagogical Value

Whether or not teachers use *Text Tanglers* will probably depend on their assessment of the pedagogical value of its activities. This reviewer found the advantages claimed for *Text Tanglers* in its documentation to be exaggerated. Letter-by-letter decoding of words, sentences, and paragraph is not a very communicative or authentic activity no matter how many software programs use this technique. There seems to be no doubt that intensive focus on a text increases learners' "exposure" to it, and for teachers who feel that the fun of working with a program as lively and flexible as *Text Tanglers* is useful and pedagogically sound, this program has some merit.

J.E.T.T. Contributor Profile

Nina Garrett is a Research Associate at the University of Illinois and a *J.E.T.T.* contributing editor. Interested readers may write to her at the following address: G70 FLB, 707 South Mathews, University of Illinois, Urbana, IL 61801.

Sequitur: An RDA/Mind Builders Software

Vital Statistics

Courseware name: *Sequitur: A Text Sequencing Tool*

Copyright: 1987

Vendor: RDA/Mind Builders, Inc.

Instructional Method: CLOZE Test

Cost: \$39.95 (single copy), \$99.95 (site license)

Copy Policy: Site license permits multiple copies in one institution; site license is transferrable

Equipment: IBM PC and Apple; color or monochrome monitors

General Description

The program *Sequitur* provides the language teacher with a structure for authoring a variation on the CLOZE test. In this case, phrases in a prose, verse, or dialogue passage are scrambled, presented three at a time, and the logically correct phrases built cumulatively into the passage until the passage is complete.

An English as a Second Language (ESL) teacher might enter the following passage about Ralph's Pretty Good Grocery from Garrison Keillor's *Lake Wobegon Days*:

In the grocery business, you to have to throw out stuff sometimes,

People bend down and peer into the meat case.

"Give me a pork loin," they say.

"One of those in the back, one of the pink ones.

"These in front are better," he says.

"They're more aged. You get better flavor."

But they want a pink one, so Ralph takes out a pink one, bites his tongue.

This is the problem with being in retail; you can't say what you think.

The program starts by giving the student the first line and some choices (all in capital letters so as not to give any external cues about the correct answer.)

In the grocery business, you have to throw out stuff sometimes

1) "GIVE ME A PORK LOIN," THEY SAY.
2) BUT RALPH IS NORWEGIAN AND IT GOES AGAINST HIS PRINCIPLES.

3) THIS IS THE PROBLEM WITH RETAIL; YOU CAN'T SAY WHAT YOU THINK.

If the student selects an incorrect answer such as #3, the negative feedback is the word "NOT" replacing the "3". Then, the prompt at the bottom of the screen reads "Press 1 or 2 (or 0 to quit)." The correct choice simply adds that text to the passage. The student goes on like this until the passage is complete. If he or she has chosen to keep score, he or she will be told that "Out of 8 lines, you found 5 lines successfully and 3 lines at the second attempt."

The Teacher's Role

The teacher must assume much of the responsibility for producing a quality activity when using this program. The manual does not give guidance about who the target audience of students is, nor does it define the goals of the activity. It gives no advice about selecting and designing the activity.

A teacher must always ask about a piece of computer software. What advantage does this program have over a paper-and-pencil text of the same type? It is also necessary to review the usefulness of CLOZE testing. In most paper-and-pencil CLOZE testing, the student writes the

missing word(s) in a statement to give it *closure*. The student is active in generating the forms rather than passively pressing 1, 2, or 3. Also, logical choices are accepted in a paper-and-pencil version, but not in *Sequitur*. In going through some of the examples provided by the authors, I felt very annoyed when I made choices that were logical yet got the "NOT" response. This might be even more annoying to an ESL student who is still trying to master the language.

The purest form of the CLOZE test does not necessarily test discrete grammar points and is more of a communicative test. *Sequitur*, however, allows no flexibility for the teacher to use interesting distractors that might be, for example, the point of a minor cultural discussion. In the above example, what if the teacher was on the line "They're more aged. You get better flavor?" and wanted the distractor 2) "The freshness date has almost expired" and 3) "They're nice and brown"? The format of this program allows only those lines that are actually lines in the passage as distractors. Moreover, by the time the student will have gotten near the end of the passage, the distractors would be lines that have already been incorporated into the previous text, giving a pretty good external cue to the right answer. Again, here the program is inferior to a paper-and-pencil version.

Teachers most often use this kind of test as a placement tool, since its results usually correlate highly with proficiency in the foreign language. It is occasionally used as a diagnostic test, when one particular kind of structure is emphasized, though it then begins to be more of a conventional discrete-point structure test. Therefore, one must seriously look beyond the surface of *Sequitur* and think about its real worth in terms of the actual classroom or lab situation.

Too Much Take, Not Enough Give

Even if this program had a certain appeal, it has some serious problems with the issue of user-friendliness. I saw glimmers of user-friendliness for the student in the clear and simple language of the introductory screen and the relatively uncluttered appearance of the passage screens. In its current version, however, starting up the program amounts to a very unclear process. The manual cover says to type MDOUBLE <cr>

but then page 1 says to type XSEQUIT <cr> or, for monochrome monitors, MSEQUIT <cr>. Primarily out of habit, I pressed the keys CTRL, ALT, and DEL (a standard way to start up any program on an IBM) with the diskette in Drive A. This got me right into the program. I tried MDOUBLE as instructed in the manual, and that did not get me into the program, whereas MSEQUIT did. Why not mention that the standard way to start up programs on the IBM will also work with *Sequitur*?

I would strongly suspect that teachers would need to be computer literate (MS DOS-fluent, in fact) in order to actually navigate in the superstructure of this program in order to create personal libraries of selections. Perhaps, I underestimate the computer savvy of most teachers. However, for all you teachers out there, here's a little test for you as potential user of *Sequitur* and creator of your own passages. If you can understand the following passage from the user manual section "Building a Library," then you go to the head of the class:

"If you look at the directory of your disk, you will see several files with the extension "SEQ". These are the text files. They are all in plain ASCII code; you can examine them by entering the command TYPE <filename>, and if you press Ctrl P, they will be copied on to the printer. They can be edited or created by using a line editor such as EDLIN or the non-document mode of a standard word-processor."

In my experience, even programmers dislike EDLIN so I can't imagine why the average teacher should be expected to know it, especially for \$40 - \$100! Is it too much to ask that this program include a very simple inputting feature?

Conclusion

Sequitur is not a computer "tool" in the proper sense of the word but rather a "game." It is not only in its title that this program runs roughshod over the partially knowledgeable consumer. In order to make it work, one would almost have to know as much about computers and program-

ming as it would take to simply write the program in BASIC oneself. I tend to suspect that many teachers using this product would find themselves requiring outside resources (e.g., more money for programmers or requests to the laboratory staff for assistance) in order to input their own texts.

For all the work required of the teacher by this program, there is still the big question about the value of such a program in the daily classroom environment. CLOZE testing, while it may be interesting, is only of use in the curriculum as a placement tool and is not often used in the process of instruction per se. Its validity as a learning

device is at best doubtful. The students who succeed at *Sequitur* may feel quite satisfied with themselves; the majority who are still struggling may suffer unintended negative effects by using it. To slightly misquote a famous Norwegian, "In the software business you have to throw out stuff sometimes."

J.E.T.T. Contributor Profile

Carolyn Fidelman, a free-lance educational media consultant and teacher, works with interactive technology at Harvard University where she is a student in the Graduate School of Education; she is a *J.E.T.T.* contributing editor. Interested readers may write to her at the following address: Department of Education, Harvard University, Cambridge, MA 02138.

Double-Up: RDA/Mind Builders Software

Vital Statistics

Courseware name: Double-Up: A Sentence Reconstruction Tool

Application: Second language learning, ESL

Instructional Method: Sentence Reconstruction Activity

Vendor: Research Design Associates, Inc.

Cost: \$39.95 for single copy; \$99.95 for school-wide license

Copy Policy: Not copy protected; may not be copied

Equipment: IBM PC

Prerequisite: A single double-sided 5 1/4" disk drive

General Description

Double-Up is a "sentence reconstruction" activity. The very basic six-page pamphlet which comes with the materials provides no statement of audience or purpose; promotional literature accompanying the package, however, states that the program is meant to supplement instruction in reading, language arts, logic, English as a Second Language, and foreign languages. The literature goes on to say that this program will teach "vocabulary, problem solving, lateral thinking, and language usage to native speaking and ESL students as well as foreign language students from grades 5 through adult." This review considers only the language teaching aspects of the program.

The program supports both a student and an authoring mode. In the student mode, the words of a short text (normally a sentence or two in length; 3 1/2 40-column lines is the maximum) are sorted into an alphabetically ordered list which the student sees displayed on the screen. The student's task is to restore the order of the original

text. To make a guess about the proper order, the student must type a pair of words. If the pair appeared together in the original text, and in the order they were typed, the pair is removed from the alphabetical list, and inserted into a growing approximation of the correct text. When enough pairs have been typed, the alphabetical list will be empty, and the original text will be completely reconstructed.

Suppose, for instance, that the original text were "I have never been to New York, but I hope to go next year." A student would see the following alphabetical list:

been but go have hope I I never

New next to to year. York,

After entering the pair "New York," the student will see the beginnings of the correct sentence and a smaller alphabetical list:

New York,

been but go have hope I I never next to to year.

At this point, the student could type in pairs such as "to New," which would move the word "to" into its appropriate position, or a pair such as "I have," which would move both words to their proper place in the sentence. To provide some continuity, the most recently moved pair is highlighted in red. Typing a wrong pair such as "been have" makes the program display error feedback to the effect that these words are not a pair; if only one or more than two words are typed in, the program indicates an error by flashing the numeral "2" in the on-screen instructions. At any time, the student can press a single key to receive help. The program then selects at random a correct pair of words and moves them

from the alphabetical list to the text. By pressing the help key repeatedly, the student can build up the entire text, or can press "S" to stop further work on the current item and see an immediate display of the correct text.

Authoring mode is a facility which allows the creation of new texts for use by students. The program consists of a very simple fill-in-the-blank style editor which allows one to type in a title and a body for each text. Since the texts reside in pure ASCII files, they can also be created using a word processor in non-document (unformatted) mode, but the *Double-Up* editor automatically observes some mandatory formatting conventions and is adequate for the purpose. For those instructors who do not wish to create their own set of items, the program comes with several pre-programmed sets: English sentences, (English) quotations, French sentences, and German sentences. The criteria for including sentences in the various sets is not documented.

Quality of Instruction

Since *Double-Up* has no presentation component, and the instructor may freely modify its language content, its value as a language learning tool must rest on the appropriateness of the "sentence reconstruction" activity as a language teaching technique. Specifically, one should know what psycholinguistic skills are developed by sentence reconstruction, and how these skills are utilized in second language performance. Unfortunately, current psycholinguistic theory cannot provide straightforward answers to these questions, and we must rely largely on intuition. Since sentence reconstruction by blind trial and error is both uninteresting and inefficient, one may suppose that the task format virtually compels students to attend to the meanings of individual words and word pairs. This in itself represents an improvement over the standard pattern drill, where items can often be processed without any reference to meaning. When evaluating the plausibility of pairs, students might also give some explicit attention to syntactic constraints (e.g., realizing that "have been" is more likely than "been have").

With respect to the range of language abilities *Double-Up* requires, the program task seems

rather similar to CLOZE activities. It should also be noted that although *Double-Up* superficially resembles "dehydrated sentence" exercises, it focuses exclusively on word order, and is probably not useful for languages with heavy inflection and flexible word order (such as Latin).

Program effectiveness will, of course, also depend on which texts a teacher provides. Most teachers will undoubtedly want texts with vocabulary and grammar related to their own syllabi, but a great many further decisions have to be made. For example, is it best to introduce grammar and vocabulary in some particular order? What is the optimum length for texts? What should the proportion of new to old vocabulary be? It would be most helpful to have some theoretical guidance on these issues, but none is provided in the documentation that accompanies *Double-Up*, presumably because none is available.

The documentation represents the sentence reconstruction task as a "puzzle," and most students are likely to proceed in the problem-solving mode, making and evaluating hypotheses about which word pairs are most plausible. The manual's suggestion that *Double-Up* functions well as a group activity would no doubt reinforce this approach. However, the puzzle aspect is not particularly stressed by the program; there is no penalty for wrong responses or help requests; the student does not even see how many tries were required to complete the reconstruction. For those students who like games and puzzles, a scoring option might lend greater motivation to the activity.

In sum, it seems plausible that the text reconstruction activity of *Double-Up* engages the student in processing the text in a meaningful way, that some language learning will result, and that the learning—since it involves meaning—will probably show up as communicative skill. Beyond this, it is difficult to say anything definite about the program's effectiveness.

Operation

The program can be used with IBM PC monochrome or color displays, in either 40 or 80 columns, but is clearly designed to look best with 40 column color. Except for the publisher's

logo screen, there are no graphics. Screen design is clean and uncluttered.

Program operation is simple. Sequencing originates at a single main menu that allows the student to choose one of the five sets of text items (including the teacher's self-authored set); once a set is selected, items are presented one after another in a fixed sequence. The student can skip at any time to the next item, return to the main menu to choose a new item set, or exit the program. The six-page manual gives a brief but adequate overview of program operation, and even young children should be able to work without problems after several minutes of use.

The most troublesome aspect of operation is the requirement that the student type in only two words at a time. Sometimes, three or more words constitute an obvious phrase, and having to type them one pair at a time—while it simplifies operation—can become quite tedious. It may also discourage students from trying to construct larger units of meaning. Though it might be technically difficult to manage, the program needs to be more flexible in this regard.

The authoring procedure for adding “libraries” (new sets of texts) to the main menu is also rather cumbersome, involving the renaming of a file according to program conventions. Moreover, editing and deleting of text items previously created appears to require direct editing of the ASCII library file. These are not technically demanding tasks, but they do require some knowledge of PC DOS, and the ideal of completely insulating the teacher from technical aspects of the program is not achieved.

More problematic is the fact that students can access the authoring process. After booting up, the program presents a page offering a choice of student or author modes. It is not possible to disable this authoring option, though once the student mode has been chosen, author mode does become inaccessible. Furthermore, pressing control-break allows a student to exit to DOS where *Double-Up* files can be edited. In light of students' well-known fondness for creatively mutilating both programs and data, this is inadequate program security.

Conclusion

Double-Up is a “sentence reconstruction” activity which engages students in meaningful processing of language. Containing no tutorial component, it is intended as an on-line supplement to classroom activity; a simple but usable authoring facility enables teachers to enter texts tailored to their own classes. The lesson has a straightforward design and is not difficult to operate. At an acceptable price of \$39.95 for the single-machine version, *Double-Up* should constitute a useful addition to a library of language courseware.

J.E.T.T. Contributor Profile

Robert Hart is associate director, language learning laboratory, at the University of Illinois-Urbana. Interested readers may write to him at the following address: G70 Foreign Languages Building, 707 S. Mathews Street, Urbana, IL 61801.

APPLIED PSYCHOLINGUISTICS

EDITORS:

Catherine E. Snow, *Harvard University*
John L. Locke, *Massachusetts General Hospital*

Applied Psycholinguistics is required reading for everyone concerned with the psychological processes involved in language. The journal gathers together the best work from a variety of fields including linguistics, psychology, psychiatry, education, language learning, sociology, speech, hearing and neurology. Articles address the nature, acquisition and impairments of language expression and comprehension. In addition, each issue contains reviews of important new books, problem-oriented reviews of important or emergent areas, short notes, discussions of previously published papers and methodological notes.

Recent articles:

The effects of early bilingual schooling on first language skills, *B. Harley, D. Hart, S. Lapkin*

Second language proficiency differences in the learning of semantically-equivalent bilingual sentences, *J.Y. Opoku*

The frequency of cohesion weakness in psychiatric syndromes, *C. Bartolucci and J. Fine*

The functions of imitation in child language, *Z. Reger*

Children's comprehension of intonation as a marker for discourse topic collaboration, *G. Holdgrafer and T.F. Campbell*

Applied Psycholinguistics (ISSN 0142-7164) is published quarterly. Subscription to Vol. 9, 1988 (US and Canada only): \$71.00 for institutions; \$36.00 for individuals; single parts \$21.00.

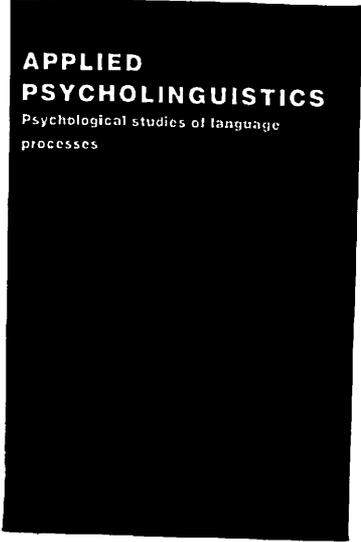
Back Volumes: 1-8 (1980-1987) \$85.00 each.

Send orders to:

Journals Department
Cambridge University Press
32 East 57th Street, New York, NY 10022, U.S.A.; or
The Edinburgh Building, Cambridge CB2 2RU, U.K.



**Cambridge
Journals**



**APPLIED
PSYCHOLINGUISTICS**
Psychological studies of language
processes

Invenntion: An RDA/Mind Builders Software

Vital Statistics

Courseware Name: Invenntion—A Logic Tool

Vendor: RDA/Mind Builders, Inc.

Instructional Method: Game or Puzzle; Reading Comprehension

Cost: \$39.95 single copy; 99.95 site license

Equipment: IBM-PC or compatible; Apple IIe

Prerequisites: Graphics card

General Description

Invenntion—A Logic Tool is a video puzzle/language game based on Venn diagrams which occur in set theory. The Venn diagram presented to the user of *Invenntion* consists of three partially overlapping colored circles. The centers of the circles form the corners of an equilateral triangle, so that the area of intersection is subdivided into four segments; the figure created in this fashion comprises a total of seven distinct areas.

The user of *Invenntion*, looking at this diagram—where each circle is labelled to represent a certain characteristic—is supposed to determine the precise distribution of a fixed number of people or objects among a given three characteristics. For example, each circle may represent a breakfast beverage—coffee, tea, orange juice—and the diagram illustrates the following: from among a group of thirsty people, a certain number drank only one thing, while some drank two (coffee *and* juice, juice *and* tea, tea *and* coffee), and others still had all three beverages.

In order to solve the puzzle, the player asks for cue statements which will help him or her determine the exact number each diagram segment represents. Typical sentences to analyze are:

—The same number of people had only juice as had only tea.

—1 more person had only tea than only coffee.

—23 people had coffee altogether.

—4 fewer people had only coffee than all three.

When the player is sure of a number from a certain segment, for example, when he or she reads “11 people altogether had both tea and juice,” he or she can select F (Fill-in) from the menu in order to type in the correct number and replace the highlighted question mark in the respective area of the diagram. When all the numbers are filled in correctly, the program will let the player know how many cues he or she needed to solve the puzzle.

Simple to Use

Invenntion is extremely easy to run; in fact, one could probably figure it out just by experimenting a few minutes with the menu without referring to the clear and concise user’s description which accompanies the package.

For people like me, however, who have a somewhat fuzzy notion of machine and hardware capabilities, it might be helpful to point out that you cannot use your run-of-the-mill office computer on which you do all of your wordprocessing (no “green screen”), but that you need a machine with a graphics card. This is perfectly logical, since *Invenntion* draws a diagram on the screen. To spare the average, educated software user frustration, the words “graphics card required” might be useful in the documentation that accompanies the package.

Is the Wide Range of Applications too Good to Be True?

In RDA’s software catalogue, the applications

of *Invenntion* are described as follows: “*Invenntion* is a program which teaches problem solving, analytical thinking, math, and language usage to native speaking and ESL students from grades 5 through adult.” Such a product description is reminiscent of the advertising style of the cosmetics industry promoting a cream as panacea for almost every woman’s beauty problems. Somewhat skeptical, I asked myself “How can one program do so many things for so many people?”

According to the creators of *Invenntion*, “[it] looks like a math and logic program...It is also a language program which tests your understanding of sentences as directly as possible.” (p. 1). They go on to say that “A language teacher would find this a rewarding means of presenting sentences which use the words *all*, *some*, *both*, *either*, *more*, and *fewer* in a way which forces learners to think about meaning.” (p. 4).

It is worth noting that the creators of *Invenntion* never clarify whether by “language teacher” they mean teacher of English as a native language or teacher of English as a Second Language. Since I am qualified as an ESL teacher, this review of *Invenntion* focuses on its possibilities as a learning tool in ESL instruction.

From my particular perspective, I confess upfront that I have certain fundamental reservations about the role of puzzles in foreign and second language learning. For most of my students, the language itself presents enough of a puzzle without making puzzles of it. In particular, it appears that the type of language usage implemented in *Invenntion*—i.e., the indirect and sometimes unnatural mode of expression necessary in order to serve the interests of the puzzle—is diametrically opposed to the common goal of language instruction, namely to enable the learners to communicate as clearly and efficiently as possible. When would a foreign speaker of English have the opportunity much less the need to say or write something to the effect of “3 more people had only coffee and juice than both tea and juice (but not coffee)?”

Granted, one could counter and point out that *Invenntion*’s primary and best applications are not in the area of teaching language pragmatics or modelling but rather in the area of reading

comprehension; after all, the creators of it state that the program will test “your understanding of sentences about as directly as possible.” This is a valid claim provided *Invenntion* is used not in the puzzle mode but in the demonstration mode, where the numbers are already filled in and the user can select S (Statement) to read a statement and to verify on the diagram that it is true; or where the user can select Q (Question) to read a question to which the answer is a number. If the number is incorrect, the computer presents a statement containing the right answer.

Demonstration mode may be an appropriate and enjoyable activity for the adolescent beginner who is only required to deal with a limited vocabulary, who needs a lot of repetition, and who can thus concentrate on the immediate meaning of the selected vocabulary items used for comparison, such as, “all, same, both, either, neither, nor, more, fewer, as, than.” Having mastered comprehension, the user may want to return to the puzzle mode for further reinforcement.

In my view, *Invenntion*’s effectiveness could be greatly enhanced for the beginner in this context if the program switched from past to present tense. Most learners would derive considerable benefit from intensive exposure to singular and plural verb endings.

In addition to the demonstration and puzzle modes, a third mode of *Invenntion*—“In your own words”—offers the possibility of creating one’s own original puzzle by replacing vocabulary that describes the initial situation with different words from the same class. However, since the program does not include a correction feature for this activity, any mistake in the choice of words will be reproduced manifold in the ensuing clue statements and may lead to a totally nonsensical result.

Whereas *Invenntion* may be an entertaining learning experience for the adolescent learner, I doubt that it would be received enthusiastically by the adult ESL learner, who is usually hard pressed to develop his linguistic abilities to a level permitting him or her to pursue a chosen field of study. Would such an adult willingly and cheerfully spend a minimum of 30 minutes at a computer only to acquire the meaning of ten odd vocabulary items? The selection at hand generally

does not represent any problems of comprehension to the often mathematically trained foreign ESL student. Since those words are embedded in a unique context only, without reference to usage or contiguous semantic fields, the mere meaning could be conveyed just as clearly and certainly more cost effectively by drawing a Venn diagram on the chalkboard and using mathematical symbols and notations for illustration.

Conclusion

It seems to me that the most promising and valuable application of *Invenntion* lies in the area of teaching math. Here, a student has unlimited practice possibilities to master solving systems of linear equations in 7 unknowns. The student could practice interpreting language representative of quantitative relationships and thus get

a true sense of what mathematics can do. Who wouldn't get a kick out of solving the puzzle with fewer cues than anyone else who relies only on mental combinations and logical reasoning?

Overall, *Invenntion* is well-designed and easily run—something sure to satisfy even the most inveterate puzzle addict. The program offers unlimited practice of a specific math problem—an exercise that in the mind of this reviewer has only marginal relevance in the broad spectrum of ESL teaching and learning.

J.E.T.T. Contributor Profile

Barbara Cooper is an experienced ESL teacher as well as a teacher of French and Italian in the Department of Romance Languages at the University of Georgia. Interested readers may write to her at the following address: Romance Languages, 109 Moore College Building, University of Georgia, Athens, Georgia 30602.

Mark-Up: An RDA Software

Vital Statistics

Courseware Name: Mark-Up

Application: Basic Punctuation Skills

Instructional Method: Drill and Practice

Vendor: RDA/Mind Builders

Cost: \$39.95 or \$99.95 Network Version

Copy Policy: Not Copy Protected

Equipment: IBM PC or compatible

PC Memory Required: Reviewer used 256K;
color or monochrome

Prerequisite: Functional English Reading

General Description

Mark-Up (MU) is a drill-and-practice program for sharpening the basic English punctuation skills. It consists of three separate, related components: expository texts on orthography; a series of exercises; and a link-up with a word processor for developing additional practice materials.

Five instructional narratives of approximately two hundred words each outline the principles of punctuation. The topics selected are capital letters and periods, commas and question marks, apostrophes, quotation marks, exclamations, and colons.

A library of fifteen paragraphs, containing some thirty words each, is available for students to practice correctly punctuating and capitalizing the brief selections. These unedited texts, stripped of all orthographic markings, appear on a practice screen.

The user-instructor may modify the practice paragraph or create additional ones via a link between MU and a word processor. Such an editing interface permits a teacher to design customized practice materials by importing American Standard Code for Information Interchange (ASCII) files as unformatted text compatible with various programs.

Skills involved in punctuation are essentially rote. They require extensive practice time, yet impose few intellectual requirements. Thus, a program promising to deliver straight forward exercises on punctuation is a most appropriate and welcome entry in the burgeoning field of computer-aided language instruction (CALL).

The MU design is elemental. MU is brief and concise in objectives, and realistic in expectation of the student-user. The basic program concept is commendable. However, due to scanty and incomplete documentation, only seasoned, old-timers in the computer stream will navigate confidently about the program to really develop custom-tailored teaching materials. Neophyte instructor-users prone to documentation phobia and unaccustomed to extraliteral interpretation of instruction are guaranteed a frustrating experience.

As the software unfolds, a main menu asks the student-user to select one of five practice categories or "levels." Subsequently, the user determines whether to activate the score-keeping option and then moves to solve orthographic puzzles. A score-keeping facility compares the number of incorrect responses against the total number of attempts and supplies a percentage score.

The Practice Screen

A practice screen is the MU nucleus. It consists of a paragraph title (optional), a brief text for editing, an inventory of punctuation, on-screen instructions/help, and a field to enter individual words. The student's typed response, ("The") appears at a point flush left immediately beneath the See/Stop line.

The directness of such a drill-and-practice approach enables even the rank beginner to intuit all the specifics of the system. However, based on observations of a number of neophyte computer users, initial entry into the program can lead to considerable confusion.

When first confronted with the practice screen, the beginner is inclined to either enter all the words requiring punctuation or to enter only those elements that appear incorrect. This is because the user manual does not sufficiently highlight the program's insistence on accepting only one word at a time. Furthermore, having to attach required punctuation to the preceding word also causes problems. Although the authors were clearly sensitive to such problems and italicized instructions in the documentation emphasize correct steps, further clarification is necessary and important because initial interaction with the program can easily determine whether or not the user continues the program.

Editing "fuzzy" passages as a form of student practice can indeed be effective. It supports the authors' contention in the documentation that MU strives to be more than a typical drill-and-practice program. Such an activity successfully underscores the function of orthography—the sharpening and clarification of the text.

When questions arise during practice, online "help" is available. The help function is subdivided into either expository texts on the principles of punctuation or single word hints related to the particular text being edited. The texts are presented as thirteen single-spaced lines explicating one of five topics on orthography. Whether appropriate breadth and depth of explanation is provided by MU is clearly debatable, though adequate detail is available to tackle the editing tasks in the exercises.

Presumably, the program is intended for a wide-ranging audience, including students learning English as a second or foreign language as well as English-speaking students. The readability of these selections is generally consistent with the expected ability of the target audience. The explication of principles, together with accompanying examples, is direct and accessible for the majority of students. The motivational value of MU exercises is not age or culturally bound; most students would benefit from them.

Feedback in Mark-Up

When students request a hint, individual random words are served up from the selection.

Such clues may result in nothing more than an increase in nonproductive practice time, since little benefit is derived from hints containing no punctuation or capitalization.

MU is not equipped with answer processing beyond a simple match routine. The program responds with a recurrent message, namely "isn't there," to each response that is not exactly equivalent to a word in the text. If the word is spelled and capitalized correctly, but is not attached to the proper attendant punctuation, MU responds with "is not right yet." Both incorrect messages are accompanied by an unambiguous computer beep which may be (thankfully) deactivated.

Single word corrections are not available. If users exhaust attempts to locate the omissions, and mistakes are still to be found, that item cannot be singled out. Users must then ask to see the entire text marked-up correctly. Two keystrokes during an editing drill summon the solution. Exiting the program is also accomplished with a two-stroke sequence.

Color has been utilized most effectively throughout the program to reinforce instructions and to signal the status of particular items being edited. Target text appears in all upper case, designated in blue. A word is immediately inserted into the target text and highlighted in red when judged correct. As the student makes the next correct insertion, the item appearing in red turns to black signifying it has become a permanent part of the corrected text. Forty column screens make the exercises visually appealing.

The student-user apparently controls time spent at each instructional level. No timed drills are included. Control over presentation rate of display material is also student determined. All texts for student practice are accessible with some minor traversing through the program. While MU is strictly linear within each level, any one of the five topics can be chosen from the main menu. However, any given level may include only the discrete punctuation skills of the preceding levels, but not of the following ones. Imposing an orthographic hierarchy does group the various skills into manageable instructional units. Nevertheless, introducing any sequence of punctuation skills and designating it as simple to difficult is not linguistically justifiable.

Built-In Editing Facility

The copyright history of MU dates its design to the early 1980s. The availability of a built-in editing facility truly distinguishes it from analogous software of that generation. The early drill-and-practice products featured invariable exercises locked away from instructors. Providing an access to drill modification expands MU's utility substantially. However, in its practical application, this feature is genuinely problematic. It involves a substantial investment of time to interpret the MU documentation and to orchestrate the text insert process with ASCII file creation.

Although creating ASCII files is not an overly complex task, each text editing-processing tool contains its own specifics and requires practice. Especially for the inexperienced, integrating generic text with the MU text input routines proves to be difficult. This process has severe formatting limitations, scanty instructions, and no examples in the documentation under the rubric "Your own text." Coupling MU with *WordPerfect* (4.1 and 4.2), *P-Edit*, the Norton Commander editor, *Microsoft Word* (4.0), and the DOS Line Editor unveiled the serious problems and limitations.

When the MU documentation directs the user to make modifications within a file called *LEVL.PUN* and *TXTnn.PUN*, it is most disconcerting to find that no such files exist. Reviewing the directory tells experienced users that the editing should occur rather in *LEVELS.PUN* and *TEXTnn.PUN*, respectively. The inexperienced computer user, however, takes every shred of documentation as immutable decree, and does not recognize typographic errors at face value. Such a user will be baffled.

Suggested Improvements

Updated versions of MU should include a modified exercise format to streamline typing each word requiring punctuation. It would be far more efficient to position a cursor on a particular location in a given text indicating a correcting, in the manner of a proofreader. Having all instructional materials, examples, and explication included in the user manual as well as incor-

porated under the online help facility would also be helpful.

The editing facility provided within MU is an absolutely essential feature. However, in its present form it is awkward and difficult to use. An addendum to MU that succinctly outlines the routines for ASCII text creation within the major word processors would be of considerable value. The instructor would not have to consult further documentation to facilitate text input. An alternative might be to include a rudimentary text editor to eliminate MU's dependence on any full-fledged word processing system.

Conclusion

As computer capabilities and computer memory increase in inverse proportions to the costs involved, CALL is assured of continual growth. The need for a wide range of instructional software is predictable. If single purpose, skill-specific language drills are to have a place in this market, they must be mechanically and pedagogically sound. Design considerations must assure underlying code impervious to minor user errors in creating customized texts. User guides must guarantee instructions and examples that are crystal clear. Documentation should be an exercise in benevolent redundancy, assuming no prerequisite computing skills or knowledge on the part of the user.

In the interests of improving the general standards of documentation, two publications should be mentioned as deserving a prominent place on the future software author's bookshelf. These are Jonathan Price, *How to Write a Computer Manual*, Menlo Park, CA: Benjamin/Cummings Publishing Co., 1984, and R. John Brockman, *Writing Better Computer User Documentation*, New York: John Wiley & Sons, 1986. MU unmistakably confirms the proposition that documentation is often the weakest link in effective CALL.

J.E.T.T. Contributor Profile

Walter V. Tuman is director of the foreign language laboratory at Louisiana State University and a J.E.T.T. contributing editor. Interested readers may write to him at the following address: Foreign Language Laboratory, Prescott 249, Louisiana State University, Baton Rouge, LA 70803.

Learning in the Workplace

Vital Statistics

Title: *Learning in the Workplace*

Author(s): Victoria J. Marsick, Editor

Publisher: Croom Helm

Publisher Address: 29 West 35th Street
New York, NY 1001

Copyright: 1987

No. of Pages: 217

Cost: \$30.00 (cloth)

Intended Users: Adult Education

Language: English

Level of Difficulty: Moderate

General Description

Learning in the Workplace is a four-part edited book that becomes the third addition to the Croom Helm series on Theory and Practice of Adult Education in North America. As in many edited texts, the individual contributions represent a variety of styles and depths of content. The transitions between the four parts as well as relationships between selections within parts are uneven. The editor's general position is that contemporary efforts to facilitate learning in the workplace have frequently faltered because old and inappropriate models of these environments prevail. The new or "emerging" paradigm recognizes that modern workplaces are better served by realizing that learners as individuals or in groups may be self-directed or prefer and respond to mentoring or coaching instruction, while traditional, behavioristic-mechanistic-hierarchical approaches fail to meet the needs of either workers or their organizations. After a semi-theoretical description of the shifting paradigm the book moves first to an unconventional examination of organizations and next to findings on workers' views of participation that are well documented in numerous other studies. The topics of self-directed learning, mentoring and coaching are addressed in Part Three. The final

section of the book contains two contributions that seem intended for two very distinct audiences—the first for the workplace researchers and the final for those who have some ability to influence the organization's responsiveness to the new paradigm.

Specifics

The editor, Victoria Marsick, introduces *Learning in the Workplace* justifying the need for another book on this topic by implying the prevailing view remains behavioristic and does not account for important changes in modern workplaces such as new understandings of decision making individuals, interactions with organizations, and worker viewpoints that are holistic. For Marsick it is important to distinguish between training, education and learning. She summarizes these distinctions by stating: "Training and education are delivery systems. By contrast, learning is the way in which individuals or groups acquire, interpret, reorganize, change or assimilate a related cluster of information, skills and feelings." She does not say directly that her interest is in workers learning the organization's culture, but her comments parallel much of the popular corporate culture literature.

According to Marsick, organizations experiencing exponential change due to needs for more highly trained workers, constant retraining, and uncertainties of technological alterations, will not be well-served by passive workers. The organization's view of the worker must be holistic, not ignoring needs for personal development, the view of workgroups must likewise be holistic, and certainly the workers themselves must be able to place themselves in the big picture. Sporadic education and training will not suffice.

For Marsick, theoretical underpinnings for her view are found in Mezirow's explanation of domains of learning and Argyris' and Schon's writing on action science. Throughout her

writing she is enamored with the metaphor of systems as holograms specifically and paradigm shifts in general analyzing events from different perspectives and different contexts, recognizing that the workplace is a dynamic system.

Among other topics illuminated in the first chapter, different types or domains of learning are defined from a brief review of the literature with an implied greater interest in what Mezirow calls dialogic and self-reflective learning than instrumental learning. Marsick seems convinced that instrumental learning for individuals has been emphasized at the expense of more social and self-reflective learning. Prior to the conclusion of the chapter, Marsick delineates eight characteristics of a paradigm for learning in the workplace that contrasts the traditional, behavioristic paradigm and three limitations to this new paradigm. The characteristics and limitations are succinct and provide a nice, clipped summary. More specifically the editor here links the more pronounced limitations with other chapters to follow.

Chapter Two by Gloria Pierce is a description of a workshop originally designed by managers to combat unionization in a corporation by helping managers with their human relationship skills. It is not an illustration of learning *in* the workplace, but instead learning *about* the workplace, since the initial attempt has been "institutionalized" into a five and a half day program evolving to serve "thousands" of people at every level of the company. The description of the content of the workshop serves as an illustration of the importance of focusing on the dialectic and self-reflective learning already described in Chapter One, although the majority of theories upon which the workshop is formulated are more strictly humanistic and precede the theories acknowledged by Marsick.

The most valuable contribution of this chapter is the report of a research impact study conducted several months after the workshop with twenty-eight of the participants to determine transfer of learning and perhaps individual perspective changes. As the workshop was described, it appeared to be rather traditional, not extremely innovative—a lot like the old fashioned sensitivity training. Researchers reported three "themes" suggesting perspective change: "self-discovery,

a move away from the external control, and holistic approaches to work." Pierce does not provide much information about the methodology for this study other than to note that the researcher conducted in depth interviews with the participants and recorded observations during the workshop in a journal. Therefore it cannot be overlooked that this impact study is based upon self-reports and the discussion of "factors that facilitate or impede learning" is contaminated by the limitations of such a methodology. The penultimate section is an exceedingly important collection of comments on "transfer to the workplace," but again it is ironic that a book whose subject is learning in the workplace describes and re-examines the pitfalls of learning outside the workplace. In the conclusion the entire focus of the chapter shifts to emphasis on the managers' dual and blurred responsibilities as educators and learners.

In building the bridge from Part One to Part Two of the book, Marsick prepares the reader to leave workshops behind and enter the real workplace world. The chapter is an unconventional, anthropological consideration of four "pre-scientific" folk model designs for organizations that might serve better to facilitate learning than existing designs. By correlating the principles of each design with standard organizational variables (goals/purposes, compensation, structure, communications, techniques/strategies, and leadership), Skruber has developed a guide which he feels has assisted individuals to understand their organization's structure and culture, their "fit" inside them, and their potential to excel. The designs and framework are useful arousing curiosity as to whether organizations desire relationships among their employees that are lateral, adult-to-adult; or vertical, adults-to-children.

Munnely's Chapter Four, "Learning Participation," is a repetitive accounting from the literature of workers' general sense of disenfranchisement in the workplace, their sense of stagnation in a hierarchical environment, and general distrust and uneasiness when their participation is solicited and encouraged. Following a skimming of research on worker participation, the chapter hones in on a comparative study of forty-eight workers in seven companies' worker participation programs. Again, there is a blur between training

and learning and the major section is a delineation of ten questions about worker education programs based on the literature and used to survey the forty-eight workers. Perhaps it is problematic that these programs are viewed as training and not learning programs. The comparative study did not support earlier findings or beliefs that workers were not wanting or willing to participate. Instead Munnely reports a keen interest in these programs and the need to advance informal and self-directed learning. She concludes with eleven recommendations for successful worker participation programs.

The attention to informal learning in the workplace is the clasp that Marsick uses to hook Part Two to the four chapters of Part Three. The first considers self-directed learning in the workplace, the next two theoretical and practical pieces on mentoring and finally a chapter on a subset of mentoring termed "coaching." The editor is perceptive in her reminders that these strategies are best suited to certain contexts and can be difficult to engage because overcoming old learning habits and passivity may be prerequisite to their effectiveness.

In Chapter Five, subsequent to a restricted review of the literature on self-directed learning, Ravid describes an Ontario study at three small manufacturing firms to determine readiness for self-directed learning of managers, professionals, support staff and blue-collar workers. Findings show that all four groups scored above average on this scale. There are several provocative speculations as to why those who had the most seniority and higher levels of education, although scoring above average on the scale, scored lowest among the four groups. Although Ravid advances the rationale for enabling self-directed learning in the workplace, caveats are not ignored, and no mention is made of any workplace sites where self-directed learning is forthrightly being scrutinized.

Bova's description of mentors and mentoring in Chapter Six is a useful recap for individuals already familiar with the concept and a worthy introduction for anyone initially exploring the mentor-mentee relationship. The concept was popularized in the early eighties and remains attractive to researchers. Many argue it's a social arrangement that must occur naturally and cannot

be arranged or forced. Thus in relation to the workplace it has been exciting to describe as it is recognized but its usefulness as a preimposed learning strategy is limited.

One workplace where a mentoring relationship may have been preimposed with positive outcomes is in elementary, junior and senior high schools. Wilcox, in Chapter Seven, describes a project in New York City where retired "master" teachers were trained to mentor new teachers who had come into the system without full college teacher training to compensate for teacher shortages. Prior to the project, resignations of new teachers were a problem. The mentoring relationship appeared to contribute to retention of the new teachers. Mentoring training involved extensive sharing and small group activities such as role playing and brainstorming. The training program workshops are outlined rather fully. Transition phases in becoming a mentor are named and anecdotal reports on the evolution of teacher-mentor relationship provided.

The final chapter in Part Three treats a strategy for learning in the workplace labelled "coaching" Again, the workplace is the school, in particular two high schools, and the workers are teachers. Coaching is differentiated from supervision because the interactions in coaching occur between peers and are not conducted by a person with formal responsibilities for performance evaluations. Like the previous chapter on a mentoring project, this chapter contains much anecdotal material that exemplifies the benefits of the strategy and underscores particular aspects of coaching that facilitate transfer of learning.

The final part of the book is titled "Implications for Research and Practice." In Chapter Nine Marsick describes the design of a study of learning in the workplace at Columbia University and Watkins University of Texas and reports on particular research issues for a study of incidental learning in organizations. Although Columbia and the University of Texas are markedly different both propose that "action science" methods are best suited to studies of the new paradigm of learning in the workplace.

The editor concludes the book by attending to practice applications, providing a neat list of Mezirow's ten "Principles of Learning in the Workplace," her own list of nine characteristics

of a learning environment, suggestions for group and individual learning plans, and the need to consider the deeper feelings of people who may become the participants in a new form of learning community.

Positive Features

Marsick and several of the other writers in this book give much needed attention to valuing the relationship between one's work life and larger life and reminding the reader that in contemporary workplaces it will be most beneficial to avoid isolating the more specific work task needs from a fuller spectrum of learner needs.

Chapters are formatted so that individual topics are titled, often capitalized, and subtopics underlined. The bibliographic references are sensibly placed at the end of each chapter rather than combined at the end of the book.

Limitations

Terms like *workers* and *workplace*, are not well-defined. Certainly there are many types of workers and workplaces. This understanding is not entirely ignored but because the terms are central to the reason for the book, expansion of their meaning would be welcomed. Although Marsick gives some justification for employers' commitment to a new paradigm of learning, her position is not extremely convincing. Organizations certainly would need stronger persuasion than is provided here related to their productivity and expansion agendas. Perhaps a chapter written on the corporate viewpoint would have filled this gap.

Marsick does observe at the close of Chapter One that the main reason workplaces exist is "productivity." It seems too critical an issue to brush aside lightly, recognizing that individual and organizational agendas may not always be harmonious. There is no case made for the new paradigm leading toward greater productivity. When one examines the current economic crisis related to our trade deficit in particular, some of Marsick's statements sound naive, e.g. "However, while emphasizing the critical importance of organizations as learning environments, a balance must be maintained between time for

learning and time for producing that for which one was hired or else the organization will go out of business." Are our most productive industries those supporting critical reflectivity of the workers? As with most works, the attempts to weave theory and practice do not always work, although the transitions between the four parts of the book are a fairly good attempt to connect these two contexts.

Some analysis of the current and projected workforce could have been provocative. Marsick writes, "Workers need more than a set of techniques; they must be able to analyze a situation to determine the nature of the problem being addressed and derive their own solutions to these problems throughout what Schon calls reflection-in-action." (p. 5) She does not explain how workers are going to learn to reflect-in-action, and we may be kidding ourselves if we think we have a generation coming along with these kinds of sophisticated learning skills. To illustrate, Ernest Boyer, President of the Carnegie Foundation for the Advancement in Teaching recently wrote:

"Democracy, if it is to succeed, requires creative, independent thinkers. Yet during our recent study of the undergraduate colleges in America, we found enormous passivity in the classroom. Students were taking notes, they were cramming for tests, and if they spoke at all in the classroom, they asked, 'Will we have this on the final examination?' That's the question that seemed to preoccupy them most of all—not learning, but recall long enough to pass the test. While we were visiting one well-known university a professor told us of a survey in which 75 per cent of the students said they could spend four years on that campus and be handed a diploma without ever saying a word in the classroom. Four years of total silence, resulting in a degree from one of the nation's most prestigious universities.... I call that passivity. The simple truth is that in too many classrooms what we are teaching is not creativity, but uniformity.... If we are teaching performance, then we are not preparing students for the world they will inherit." (p. 5)

College students are obviously not the only workers-on-the-way. An interim report on the

School-to-Work Transition "The Forgotten Half: Non-College Youth in America" (January 1988, The William T. Grant Foundation Commission on Work, Family and Citizenship) contains findings that may also challenge the feasibility of the new paradigm. The difficulty of helping the paradigm to shift may be exceedingly more complex than assumed when the educational attainment and preparation of the projected workforce is assessed.

Marsick's writing about the humanistic model is fuzzy: "Subjective feelings are highly important as indicated by their 'touchy-feely' label, whereas behaviorists espouse the opposite end of the same continuum" (p. 2). What does she mean?

Finally, the tiny typeface is exceedingly disturbing to read.

Conclusion

Learning in the Workplace is a pastiche of theoretical and practical thinking that challenges past and present views of worker education, training, and learning. Its greatest flaws are the omissions of profiles of the projected workforce (especially considerations of preparation for informal and self-directed learning) and the perspectives of workplace leaders and executives regarding learning in another paradigm. The general impact of this book will be mostly for those who think about workplace change. With the exception of a couple of chapters in Part Three, there is little here for the doers.

J.E.T.T. Contributor Profile

Margeret E. Holt is an associate professor of adult education at the University of Georgia and an associate with the Charles F. Kettering Foundation. Persons wishing to write to her may do so at the following address: Adult Education Department, Room 418 Tucker Hall, University of Georgia, Athens, GA 30602, U.S.A.