

5-1-1992

A guide to computer-based training for the graphic arts industry

David E. Pease

Follow this and additional works at: <http://scholarworks.rit.edu/theses>

Recommended Citation

Pease, David E., "A guide to computer-based training for the graphic arts industry" (1992). Thesis. Rochester Institute of Technology. Accessed from

This Thesis is brought to you for free and open access by the Thesis/Dissertation Collections at RIT Scholar Works. It has been accepted for inclusion in Theses by an authorized administrator of RIT Scholar Works. For more information, please contact ritscholarworks@rit.edu.

School of Printing Management and Sciences
Rochester Institute of Technology
Rochester, New York

Certificate of Approval

Master's Thesis

This is to certify that the Master's Thesis of

David E. Pease

With a major in Graphic Arts Publishing
has been approved by the Thesis Committee as satisfactory
for the thesis requirement for the Master of Science degree
at the convocation of

May, 1992

Date

Thesis Committee:

Thesis Advisor

Graduate Program Coordinator

Director or Designate

**A Guide to Computer-based Training
for the Graphic Arts Industry**

by

David E. Pease

A project submitted in partial fulfillment of the
requirements for the degree of Master of Science in the
School of Printing Management and Sciences in the College
of Graphic Arts and Photography of the
Rochester Institute of Technology

May 1992

Thesis Advisor: Professor Frank Cost

A Guide to Computer-based Training

for the Graphic Arts Industry

By David E. Pease

Executive Summary

Industry in the United States is facing one of the greatest hurdles of all time, that of retraining its work force. The printing industry, in particular, faces a continuing lack of experienced and knowledgeable people. It has begun cooperative efforts to retrain workers in the basic skills needed to run a printing operation. Subjects like basic math, accounting, writing, and sciences round out the curriculum. Also on the scene are entrepreneurial training firms. Off-site training typically goes for \$250-350 per person per day; the on-site instruction goes for \$400 and up. And then there are options like computer-based tutorials. A firm in San Francisco ships an animated lesson on chokes and spreads to desktop publishers for \$20. The demand for this packaging of instructional material has been high. But is it effective?

Ever since computers arrived in businesses and institutions of higher learning, people have been trying to assess just how helpful they really are. In measuring the instructional finesse of computers in the classroom, there is no exception. Exhaustive studies have been done around the world, each reaching debatable conclusions. So there has been a toss-up in academic circles and in the media—"Do computers enhance learning? If so, how much?"

Researchers received grants and went to work, quantifying the effects of computer-based instruction on everyone from grade-school children to college students. After statistically examining hundreds of individual, controlled studies, they reached some meaningful conclusions in support of computer-based teaching. In general, achievement of the average student did not increase dramatically as a result of computer-based teaching. Students in the 50th percentile on exams reached the 65th, or near to it. Researchers have found, however, that the use of computer-based instruction reduces the time needed for learning, usually by a factor of 33 percent. Other benefits realized are improved student attitude towards the material to be learned and towards computers in general.

Computer-based training has the potential to lower training costs by reducing the time needed for instruction. It is also easier and less expensive to mail a disk to remote locations than it is an industrial trainer. So why not hop on the bandwagon as it just starts to roll? Multimedia computers and great software packages are on the market, ready to be used for training applications and even presentations.

Although these possibilities are now becoming reality across the country, most firms do not have the programming or teaching expertise to build computer-based training modules in-house. Unless a firm has the equivalent of a graphic designer, a programmer, an instructional designer, extra time and extra money, it's not ready to take the plunge into development.

A little extra knowledge can go a long way when it comes to the acceptance of a new medium that could positively impact the printing industry. The following pages explain concepts, models, and basic ingredients of computer-based training. While some members of the graphic arts community regret the onslaught of computers in their industry, the next phase—instruction—has the potential to enhance worker morale, improve quality, and increase profitability during the next decade.

Contents

1.0	Training needs and solutions	1
1.1	Current state of training in the printing industry	1
1.2	An interactive approach	2
1.3	Time and money	3
1.4	Future applications of computer-based training	4
2.0	Effectiveness of computer-based training	6
2.1	Quantifying benefits and returns	6
2.2	Meta-analysis	7
2.3	Study results	8
2.4	CBT and college-level students	11
3.0	Essential Elements of CBT	13
3.1	Examining dialog	13
3.2	Student response	15
4.0	Types of CBT	17
4.1	Tutorials	17
4.2	Drills	18
4.3	Simulations	19
4.4	Games	21
5.0	Producing computer-based training	22
5.1	A team effort	23
5.2	Eight steps to follow	23
5.3	Harvard Law School example	24
5.4	Preliminary evaluation	25
6.0	Software and hardware requirements	27
6.1	Authoring software	27
6.2	Macintosh-based software	28
6.3	DOS-based software	29
6.4	Hardware requirements	30
6.5	Pre-configured solutions	31
7.0	Empowered, but enabled?	33
8.0	References and sources cited	34
	Appendix I: Multimedia training resource guide	36
	Appendix II: Interface guidelines	38
	Appendix III: Creating “Platemaking Fundamentals”	43

1.0 Training needs and solutions

It appears that nearly all has been said and little has been done about the current dearth of training programs implemented in the American graphic arts industry. With increased competition from abroad and rapidly changing printing and pre-press technologies, now is not the time for a short-range strategy for doing business. The opportunity now exists to invest in the work force. Technology alone is not enough to guarantee efficiency and quality in the future.

Consider the following:

“According to a 1989 survey of commercial offset printers conducted by the National Assn. of Printers and Lithographers (NAPL) and *American Printer* magazine, more than one-third of 800 respondents believe that the skills problems of the graphic arts industry are serious or very serious; 74 percent believe they have a basic skills problem; and 63 percent could point to a specific negative event occurring in their plants attributable to poor basic skills.

Add these grim findings to demographic studies indicating that 75 percent of the people who will be in the entire U.S. work force in the year 2000 are already working, and the need to correct this serious problem, and soon, becomes even more apparent” (Ferris 1991).

1.1 Current state of training in the printing industry

The NAPL’s 1991 report, *The State of Training Programs in the Printing Industry*, concludes saying “While some printers are extensively involved in training, overall, the graphic arts is still an industry in which a minimum of training is provided at a minimum of cost” (Arajas 1991). It appears that with the onset of cost-cutting, especially in these tough economic times, training considerations are the first thing to go. A better educated and dedicated work force may be spoken of as a necessity for doing business in the 90s, but the means of getting there, time and money spent re-educating employees, is viewed as an expendable luxury. Even if a company does implement training programs, chances are slim that the company realizes the benefits: “Only 4 percent of those surveyed [by the NAPL] use performance measurements of some kind to evaluate the effectiveness of training and basic skills programs” (Arajas 1991).

Already industrial trainers, consultants and software experts have arrived on the scene to meet the instructional needs of the printing and publishing industry. Off-site training programs typically go for \$250-350 per person per day. On-site tutoring costs \$400 and up per day (Cohen and Moonitz 1991). Yes, the expense of training is high, but the longer-term costs of doing without it are even greater.

Arriving at a more communal solution, the NAPL has recently introduced its Carl Didde WorkPLACE to address worker deficiencies in the printing industry. The four-part program tackles on-the-job math computations, communications, graphic arts processes, and critical thinking/problem solving. The program has been well-received so far, but as we all know, classroom-style instruction isn't for everyone. Training videos are being produced by other firms to address some of these needs.

1.2 An interactive approach

It may only be a matter of time before industrial trainers team graphic arts companies' Macintoshes and PCs with instructive curricula to better show physical, chemical and even mathematical processes and procedures. Instead of delivering lessons in a classroom setting, some companies are letting workers "discover" tough concepts themselves with a little help from pre-programmed lessons and interactive simulations, tutorials, games, drills, and tests made for the computer. Those in the field call this computer-based training (CBT).

Xerox recently created an interactive game to teach customers, technicians and others how to use a new copier, according to instructional designer Michael Yacci, Ph.D., of RIT. The benefits of this game to Xerox include efficiency in teaching. The company can tutor many people in widespread locations how to use a given machine, but the lesson only has to be created and delivered once. The costs associated with this type of training decline the longer a system is in use and the more people it addresses.

With CBT, students don't necessarily learn lessons better. They may, however, learn lessons in less time and using fewer resources. These economies are documented in other industries—airlines' flight simulators are the most obvious example. Crashing a simulator is much less expensive than destroying a 727, for instance. More takeoffs and landings

may be practiced. Flight conditions can be controlled and changed one variable at a time to focus on problem areas.

These instructional technologies have been largely proprietary up to now. Wealthy companies have had the capital and expertise to hone their own workers' skills, adding to a collective competitive advantage. Perhaps companies willing to contribute to programs such as the Carl Didde WorkPLACE will find value in supplementary CBT modules that could also be industry association initiatives benefiting many companies. Costs, of course, would be spread among association members, reducing the per person expense of training.

1.3 Time and money

Computer-based training allows for better-than-real world learning conditions. With CBT training, photo polymerization could be simulated in slow motion, ink/water emulsions could be studied at the molecular level, estimating scenarios could be played without aggravating clients, and web tension could be regulated without starting the press. CBT makes the invisible visible, slows and accelerates time, and allows for the immediate isolation of variables in complex systems. But it all costs money and takes time to create.

Xerox's instructional game, previously described, took about 1,300 hours of teamwork to create one hour's worth of game. It was a very complex game at the computer code level, according to Yacci. Average computer-based training applications emerge at the rate of 400 development hours to one hour of finished product. Of course, the money spent varies from application to application.

The good news is that new desktop computers and software are making it possible to create quality interactive training modules, using multimedia tools, at the home office. Microsoft Corporation's new MPC standard for incorporating sound, CD-ROM, and video and Apple Computer's Quicktime system extension bring far-reaching capabilities to the masses for relatively little money.

"Often when we think of [multimedia], we imagine brightly colored geometric shapes and text whizzing across computer screens to the sound of canned music—eye candy that ultimately leaves us hungry for more information. But the promise of multimedia is more than that. In addition to combining sound, video, graphics, animation, and text into an

electronic document, true multimedia implementations demand participation on the part of the operator. That interactivity, or relationship between [user] and computer, separates multimedia from movies and MTV. It enables us to learn and gather information through natural, non-linear paths, making multimedia technology an exceptional communication tool” (Frenkel 1991).

Pixel Ink Consultants in San Francisco recently embarked on its first multimedia journey to address the pre-press training needs of printers and service bureaus who desire cost-effective tutoring of their clients. At \$20 per disk, Pixel Ink provides a "Spreads & Chokes" animated tutorial that plays on any Macintosh. The program occupies 1.7 MB (compressed) on a standard 800K floppy. Pixel Ink Consultants, a two-man team of pre-press expertise, and a computer graphics animator produced the lesson in-house. According to one of the partners, Hans Hartman, their project has been well-received by both industry and the press. After several magazine mentions this winter, Pixel Ink has been selling about 30 disks per day.

1.4 Future applications of computer-based training

Interactive, computer-based training could have a legitimate future in the graphic arts industry, which should experience increasing automation in the near future. “Training operators of complex, predominantly automated dynamic systems [as in pressroom machinery] is a challenging problem. The risks and complexity of such systems limit the effectiveness of on-the-job training; conventional classroom instruction with printed and audio-visual media fails to capture the essential elements required to supervise a complex system in real time. There is a growing sense that computer-based intelligent tutoring systems (ITS) offer a much needed alternative learning environment. Ideally, an ITS provides one-on-one instruction, apprenticeship-style learning (without the risks of training on the actual system), and greater tolerance for error styles (Mitchell and Govindaraj 1991).

Before any team of software developers, instructional designers, animators, graphic designers, and writers tackle such a project, the printing industry is going to want to know what it’s about to enter. Furthermore, ambitious industrial trainers in the graphic arts industry should know the efficiencies of learning, the elements of producing, the costs of

creating, and sources of information for interactive, computer-based modules before diving headlong into a project.

2.0 Effectiveness of computer-based training

Ever since the late 1950s when computers started being used in any kind of training or in a classroom learning environment, educators have been assessing the merits of these systems. Educators and researchers around the globe have written hundreds of reports and studies on the merits and shortcomings of computer-based instruction, each with its own experimental conditions. Since 1965, the U.S. Office of Education and the National Science Foundation have funded hundreds of computer projects in education. The Exxon Foundation, the Sloan Foundation, other foundations and numerous computer and software companies have poured millions into the development of computer-based teaching and learning (Kulik, Kulik, and Cohen 1980).

In some cases, the use of computer-based teaching has been vehemently resisted. “The reluctance of teachers to use technological devices is due at least in part to uncertainty about the effects of technology on learners. Opponents of technology argue that it stifles creativity, rewards conventionality, and encourages rote performance. Advocates, on the other hand, argue that technological devices free learners from lockstep instruction. They claim that technology can help learners move at their own rates through curricula suited to their own needs” (Kulik, Kulik, and Cohen 1980).

2.1 Quantifying benefits and returns

It has been difficult for anyone to say, conclusively, exactly what benefits should be expected from a computer system that teaches students. There still is no standard methodology when it comes to evaluating a teaching system. Each sponsoring organization has its own objectives when it comes to funding computer-based instruction projects. But there have been enough studies measuring the same results to make general statements about computer-based instruction. This has enabled researchers in the field like James Kulik, Chen-Lin C. Kulik, and Peter Cohen to embark on a series of in-depth research projects, referred to as *meta-analyses*, which quantify benefits of these teaching and training tools. Some of these important meta-analyses have been made possible through grants by the National Science Foundation.

After reading through hundreds of pages of statistical analysis and experimental proceedings, the reader will probably be convinced that instructional technology used in the

classroom or in a training situation has certain measurable benefits. The first is that computer-based instruction does not make an earth-shaking difference in achievement compared to normal classroom instruction when students are tested on material and then retested. Often, there is an improvement with a test group that has participated in computer-based instruction, but it is statistically small—usually less than 0.3 standard deviation units. Secondly, computer-based instruction significantly reduces the time needed for learning. Lastly, computers utilized in teaching have a small but positive effect on the attitudes of students toward the subject matter they are studying.

There are three studies worth examining if quantitative and qualitative proof of the benefits of computer-based instruction is necessary to push forward a CBT project. Actually, there are many involved studies that would suffice as credible documentation (which are noted in these studies' references), but these three are a good place to start; they might fulfill most people's needs. The meta-analyses are as follows: "Effects of Computer-based Teaching on Secondary School Students," Kulik, Bangert, & Williams, *Journal of Educational Psychology*, 75: 19-26, 1983; "Instructional Technology and College Teaching," Kulik, Kulik & Cohen, *Teaching of Psychology*, 7: 199-205, 1980; and "Effectiveness of Computer-based College Teaching: A Meta-analysis of Findings," Kulik, Kulik & Cohen, *Review of Educational Research*, 50: 525-544, 1980.

2.2 Meta-analysis

Meta-analysis is, essentially, the analysis of analyses. The term was first used by G. Glass in 1976 to describe "the statistical analysis of a large collection of results from individual studies for the purpose of integrating findings. Researchers carrying out a meta-analysis locate studies of an issue by clearly specified procedures. They characterize features of the studies and study outcomes in quantitative or semiquantitative ways. Finally, meta-analysts use multivariate techniques to describe findings and relate characteristics of the studies to outcomes" (Kulik, Kulik and Cohen 1980).

"When studies of an issue run into the hundreds, and findings are diverse, reviewers often see what they wish in the collected results. The use of quantitative methods and statistical tools somewhat constrains a reviewer's fancy. By applying to a collection of results the same objective methods that researchers use in analyzing results from an individual study,

the meta-analyst is able to draw reliable, reproducible, and general conclusions” (Kulik, Kulik, and Cohen 1980).

2.3 Study results

In the meta-analysis of studies performed on secondary school students, researchers hypothesized that while it had been established that computer-based instruction significantly aided lower grade level students in subjects like mathematics, it may actually get in the way of upper-level students. “Kulik (J.) suggested that at the lower levels of instruction, learners need the stimulation and guidance provided by a highly reactive teaching medium. At the upper levels of instruction, a highly reactive instructional medium may be unnecessary. College learners apparently profit from working by themselves on problems before receiving individual evaluations and prescriptions for further work” (Kulik, Bangert, and Williams 1983).

Why set up a pressroom supervisor on a problem-solving computer tutorial when a manual, workbook, and conventional classroom instruction would be more effective? Should equipment manufacturers for the graphic arts industry, or any industry, further pursue interactive training programs for clients’ computers?

Kulik, Bangert and Williams selected 51 studies for final examination performance, performance on retention examinations, attitude toward subject matter taught in the experiment, attitude toward computers, attitude toward instruction, and time needed to learn. To quantify findings of the many studies these researchers used Effect Size (ES), defined as “the difference between the means of two groups divided by the standard deviation of the control group (non-computer-based instruction),” for comparison.

“In 39 of the 48 studies with results from final examinations, students from the computer-based instruction (CBI) class received the better examination scores; in 9 other studies, students from the conventional class got the better scores. A total of 25 of the studies reported a statistically significant difference in results from the teaching approaches. Results of 23 of these studies favored CBI, and the results of 2 studies favored conventional instruction. . . By using the index of effect size (ES), we were able to describe the influence of CBI with greater precision. The average ES implies that in a typical class, performance of CBI students was raised by 0.32 standard deviations. . .

Using this guideline, we see that students from the CBI classes performed at the 63rd percentile on their examinations whereas students who only received conventional instruction performed at the 50th percentile on the same examinations. Or put another way, 63 percent of the students from CBI classes outperformed the average student from the control classes” (Kulik, Bangert, and Williams 1983).

Computer-based instruction excels at reducing the amount of time needed to learn subject matter. In the given meta-analysis of secondary school students, only two studies included comparative data on the amount of time needed for students to learn. “In one of the studies, students spent 135 minutes on instruction and study when taught with computers and 220 minutes when taught in a conventional manner” (Kulik, Bangert, and Williams 1983). This represents a 39% savings in time. In the other study, students spent 90 minutes on instruction and study with CBI and 745 minutes when taught conventionally. The 88% time savings is great, but may be considered closer to an extreme case.

Other comparisons included information on the attitudes of students toward the course material during the time of the experiment, toward the method of instruction and towards computers. In general, students in the CBI test group had more positive feelings toward the subject matter, computers and quality of instruction. It should be noted that there was no statistically distinct difference between students in the control (conventionally taught) group and the CBI students in their attitudes towards the quality of instruction.

Another facet of this same analysis compared years in which studies were conducted. The most recent studies showed stronger positive effects of computer-based instruction. Other analyses have shown similar results. These researchers believe that it is unlikely that the stronger results can be attributed to a switch in later years to better research designs. They say that it is more likely that instructional technology has been used more effectively in later years.

Perhaps the above findings shed some light on possible benefits of computer based instruction for individuals who have reached a secondary school-level of education. But what about college-educated employees? Do the numbers change for those who have attained a higher level intellectually? Would they benefit from CBT to the same extent? These questions might be asked by those planning on producing CBT focusing on tone

reproduction theory or “chokes and spreads” for, say, college-educated graphic designers or photographers.

2.4 CBT and college-level students

James Kulik, Chen-Lin Kulik and Peter Cohen published a meta-analysis “Effectiveness of Computer-based College Teaching...” in 1980 at the University of Michigan. As in the analysis of secondary school studies, CBT showed small but positive gains in student achievement on final examinations. Computer-based training was responsible, again, for significant savings in time spent learning. Finally, the analysis showed little correlation between findings and experimental design, study setting, and manner and date of the publication of individual studies included.

The researchers narrowed their field of studies to 59. Included in these studies were four different kinds of CBT: tutoring, computer-managed teaching, simulation, and programming the computer to solve problems. Types of CBT will be explored more fully in Section IV. “The computer instruction also varied along two other dimensions. In some studies, the computer substituted for conventional teaching—replacing lecturers, recitation sections, conventional readings, problem assignments, or some combination of these” (Kulik, Kulik, and Cohen 1980).

Another interesting concern in this study was the relationship between aptitude and achievement in a class. In a conventional class setting, all students receive the same lecture, materials, and overall instruction time. Here, student aptitude has a large effect on achievement when it comes to the final exam. Some students are simply more adept at absorbing and understanding information than others. When there is a time constraint element to learning (class time), the “quicker” students perform better on final examinations. But in computer-based instruction, students receive varying time allotments of instruction; they use as much time as they need to learn the material. In this type of instructional environment, student aptitude plays less of a role in final performance. Thus in an industrial training situation, CBT could compensate for classroom aptitudinal differences among students.

A total of 54 of the 59 studies measured student achievement. “In 37 of the 54 studies, CBI [computer-based instruction] examination performance was superior to examination performance in a conventional class; 17 studies favored conventional instruction. Fourteen of the 54 comparisons reported statistically significant differences between teaching methods. Results of 13 of these studies favored CBI, and results of one study favored

conventional instruction. If no overall generalization about the effect of CBI was possible, one would expect about half of the cases to favor CBI and half to favor conventional teaching. Instead, a clear majority of studies favored CBI. We were therefore able to reject the null hypothesis of no effect of CBI on student achievement.” Cohen regards the effect of CBI as small. Again, CBI-taught classes showed average final scores (on the same exams as the control group’s) that were 0.2 standard deviation units higher. “A typical student in a CBI class was performing at the 60th percentile on examinations, whereas the typical control student performed at the 50th percentile” (Kulik, Kulik & Cohen 1980).

Eight studies collected data for time spent learning in CBI and conventional classes. Each of the studies found that the computer-based class realized substantial savings in time. In all cases, the results were statistically significant. “On the average, the conventional approach required 3.5 hours of instructional time per week, and the computer-based approach required about 2.25 hours. This is a substantial and highly significant difference between methods. There appear to be little doubt that students can be taught with computers in less time than with conventional methods of college teaching” (Kulik, Kulik, and Cohen 1980).

Eleven of the 59 studies reported news on student attitudes toward instruction, either computer-based or conventional. CBI ratings were higher in eight of the studies; conventionally-taught classes were judged more favorably in the remaining three. In general, CBI classes received slightly higher ratings than conventional ones. The effect of CBI versus conventional for students’ attitudes toward subject matter taught during the experiment was also very small in these college classes. But again, students ranked the CBI classes higher.

3.0 Essential elements of CBT

The computer-based training field is relatively deep. Those who have spent their lives reading graphic arts technical literature might find themselves overwhelmed with a new universe of information on the computer interface, the student/teacher dialog, and even the basic components of communication or interaction. There's quite a bit to think about; the aim here is to provide examples of what the CBT experts provide as fundamental types of computer interaction and point the way to further reading.

William E. Montague of the U.S. Navy Personnel Research and Development Center said that "planning to use computers for instruction is secondary to planning or designing a course of instruction in a broader sense" (Montague 1988). The incorporation of computer instruction in training is not going to make a humdrum course of instruction into something extraordinary. Good material and instructors dictate where computer courseware can enhance the learning experience.

"The primary idea is that the instructional environment must represent to the learner the context of the environment in which what is learned will or could be used. Knowledge learned will then be appropriate for use and students learn to think and act in appropriate ways. Transfer [of knowledge] should be direct and strong" (Montague 1988).

As one would expect, understanding the learning environment and developing a method of delivering information so that a student can master it is the toughest job when designing computer-based courseware. One might say that the courseware designer stands to learn more from the process of creating than the students who will benefit from it. "Competent performance of complicated tasks develops slowly, and requires an environment that supports the development of sub skills, knowledge and coordination. It requires analysis of competence or skill and breaking it into manageable and appropriate chunks for students to learn. Although utilizing computer-based technologies in this effort can help both design of instruction and its delivery, it can provide additional difficulty" (Montague 1988).

3.1 Examining dialog

When considering human-computer interaction, one might imagine the typical model of a dialog—an ideal Socratic state where the computer mentor guides and challenges the learner

to a new level of understanding. In this dialog, exchanges of information take place; listening is just as important as speaking. Each participant carefully considers what the other has said before replying. A long-winded speaker detracts from the frequent interchange of ideas, and thus hampers interactivity and learning. Socrates' ideal state has been admired for centuries, and it is the direction courseware designers strive to reach in their development work. There are few truly great human teachers who keep all students actively involved through questioning and challenging weak understandings; computer-based training has proved successful by doing just that.

“As we have been faced, for thousands of years, with the difficult problem of educating more and more students, we have used less interactive forms of learning, such as the lecture and the book. The learner, today, is primarily a spectator in our formal learning institutions. In a room with 25 to 30 students, only an exceptional teacher can offer fully individualized attention. So we do not react to individual differences in most cases” (Bork 1987).

“We cannot reasonably expect to have enough good teachers to conduct all education interactively. But we can, with computers, amplify the effect of the few teachers with great skills in this direction by persuading them to develop technology-based learning material, assuming we can provide them with a reasonable development environment. In direct conversation, these teachers can reach few students, but through the surrogate of the computer they can, potentially, reach vast worldwide audiences” (Bork 1987).

Newcomers to the world of computer-based instruction are mostly awed and entertained by the interaction of teaching modules. Because of the dazzle involved, it is easy to miss the thought processes involved in making effective courseware, and relatively inexperienced developers are satisfied with less than satisfactory results when it comes to the interactive level of the project. Thus, there is a distinct danger of producing ineffective courseware that looks good. Guidelines for field testing a computer interface will be provided in section five.

“Two numerical factors affect degree of interaction. A critical factor is the average time between interactions, for all the users of the computer dialogue. The figure that we strive for at the Educational Technology Centre at Irvine is an interaction time of 15 to 20

seconds. This time is measured as follows: the clock starts when the student presses return or enter [or clicks a mouse], after having answered a question, and the clock stops when the student begins to think about how to respond to the next query. Designers may choose longer or shorter times, depending on the pedagogical intent. And the average time may differ considerably from student to student” (Bork 1987). This timing measurement sets the tempo of the conversation between computer and learner. It is important to keep the student actively engaged with a suitable pace; it is also important not to go too fast or else the dialogue may feel more like a rapid-fire interrogation. Thinking about and observing conversations may yield further insight into how new developers would want their interactive teaching systems to work.

3.2 Student response

A second measure in interactive, computer-based instruction is the time for student response. “Long times correspond to a non-interactive situation. Some computer dialogues ‘watch the clock’ and attempt to stimulate student response when nothing has happened for some time. Short times are often not desirable in a learning environment because we want to encourage thinking as part of the learning process. When a student replies to a question, we want the student to think about the question, not just react. In some places the developers may even want to discourage a quick response, because too little thought time has gone into the process. In other situations, such as building student intuition, a rapid response is desirable” (Bork 1987).

In graphic user interfaces, such as Microsoft’s Windows or Apple’s Macintosh Operating System, it is often preferable to create software that matches the operating system’s simplicity. Developers usually create quite clever interfaces for things like HyperCard stacks that require little more of the user than “clicking” and “dragging.” Decision, or dialogue, boxes are almost always multiple choice interactions. The user need only double-click on the line that corresponds to the desired choice. Bork cautions instructional designers of computer courseware regarding the use of such slick interfaces when it comes to teaching materials. Here, the quality of student response is more important.

“The best type of input is the student’s everyday language. It is not difficult, with a reasonable design process and with simple computer techniques, to build programs that do a remarkably good job of responding to typical student inputs in everyday language. . .

Designers specify the likely student responses to the questions. Simple string matching, with some logic, is usually sufficient. We have found it better to match on beginning fragments of words, rather than whole words, to overcome some typing and spelling problems. When the material is used with students, in the formative evaluation process, new possibilities that the designers overlooked will come to light, and the material can be improved and tested again. A particularly poor form of input is multiple choice. Multiple guess decisions, as they are described by students, seldom correspond to the real world. In most important decisions, there are far more than four or five possibilities, so multiple choice questions seldom test for the situations that the students encounter in life” (Bork 1987).

Before proceeding to examples of different types of CBT, there are a few guidelines published that are worth mentioning. Montague calls them “Heuristics for vocational education,” which is an distillation of work published in 1925. They are the following: “Replicate the working environment in training (same operations, same tools, same machines, same thinking and manipulations); adapt training to trainee (assess and consider intelligence, aptitudes, interests); give frequent practice (enough to develop skill); use ‘expert’ trainers who know the job (to design training and to teach)” (Montague 1988).

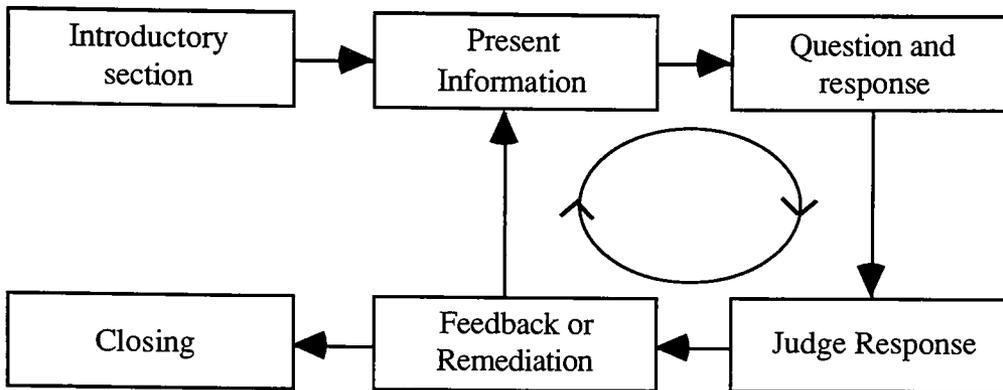
4.0 Types of CBT

An excellent compilation of fundamental information on CBT was published by Stephen M. Alessi and Stanley R. Trollip. The book, which is used as a text for graduate students of instructional technology at Rochester Institute of Technology, is *Computer-Based Instruction: Methods and Development*. A second edition is now available through the publisher, Prentice Hall. A collection of introductory excerpts on computer-based tutorials, drills, simulations, and games follows. These choices promise to have straightforward applications for those hoping to develop materials for the graphic arts industry.

Alessi and Trollip state that “for instruction to be effective, it is necessary that the following phases be present: presenting information, guiding the student, practicing, and assessing student learning” (Alessi and Trollip 1985). Tutorials, they say, incorporate the first two previously mentioned phases. Tutorials assume the role of the instructor throughout CBT.

4.1 Tutorials

“Tutorials are used in almost every subject area from the humanities to the social and physical sciences. They are appropriate for presenting factual information, for learning rules and principles, or for learning problem-solving strategies. A tutorial begins with an introductory section that informs the student of the purpose and nature of the lesson. After that a cycle begins. Information is presented and elaborated. A question is asked that the student must answer. The program judges the response to assess student comprehension, and the student is given feedback to improve comprehension and future performance. At the end of each iteration, the program makes a sequencing decision to determine what information should be treated during the next iteration.



Alessi & Trollip 1985, 68

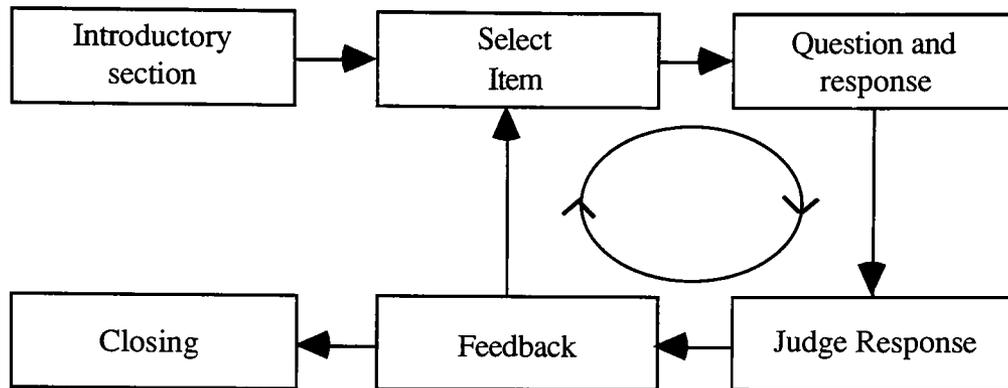
The cycle continues until the lesson is terminated by either the student or the program. At that point, which we call the closing, there may be a summary or closing remarks...Not all tutorials engage in all these activities. However, we contend that an effective tutorial will include all components” (Alessi and Trollip 1985).

4.2 Drills

Drills are used primarily for Alessi and Trollip’s third aspect of the instructional process, practicing. “The practice phase of instruction is very important, and drills, in combination with tutorials and other methodologies, are necessary for learning information in which fluency is required, such as basic math skills, foreign languages, spelling, usage and vocabulary.

Drills are not intended to teach. The problem arises when teachers assume a drill is capable of teaching new information and use it as if it should. Drills must be preceded by instructional methodologies that present the information and guide the student through initial learning. In computer-based instruction this might mean preceding the drill with an appropriate tutorial or simulation. It might also mean preceding the computer-based drill with reading in a textbook, a classroom lesson, or a group discussion.

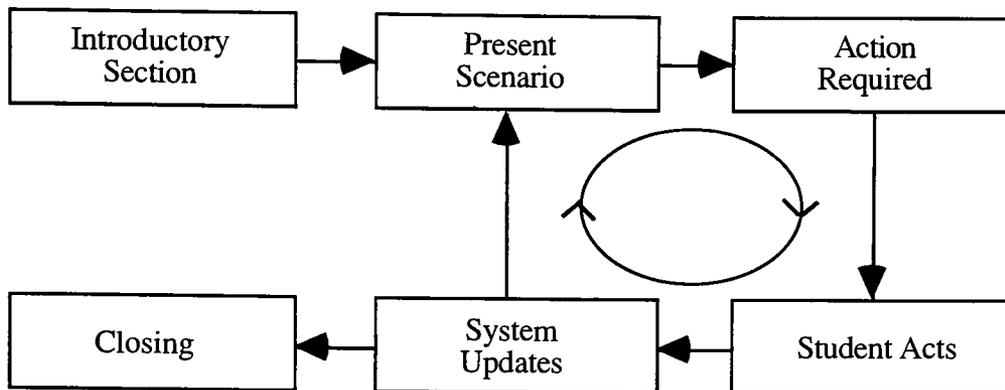
In a drill, an item is selected. The item is displayed. The student responds; the program judges the response. Finally, the student receives feedback about the response” (Alessi and Trollip 1985).



Alessi & Trollip 1985, 135

4.3 Simulations

Simulations may be used for any of the four previously mentioned phases—initial presentation of information, guiding the learner, practice, assessing learning, or any combination of these. “In the educational context, a simulation is a powerful technique that teaches about some aspect of the world by imitating or replicating it. Students are not only motivated by simulations but also learn by interacting with them in a manner similar to the way they would react in real situations. In almost every instance, a simulation also simplifies reality by omitting or changing details. In this simplified world, the student solves problems, learns procedures, comes to understand the characteristics of phenomena and how to control them, or learns what actions to take in different situations. [A web break, or plate blinding, for example] In each case, the purpose is to help the student build a useful mental model of part of the world, and to provide an opportunity to test it safely and efficiently” (Alessi & Trollip 1985).



Alessi & Trollip 1985, 176

Simulations can be classified as being either physical in nature, procedural, situational, or process-oriented. Graphic arts training professionals are probably most interested in simulations of the physical or procedural sort. “In a computer-based physical simulation, a physical object is displayed on the screen, giving the student the opportunity to use it or learn about it. Typical examples are a machine that the student must learn to operate or some scientific laboratory equipment to be used in an experiment” (Alessi and Trollip 1985).

“The purpose of most procedural simulations is to teach a sequence of actions that constitute a procedure. Common examples are operating a hand-held calculator or telephone, performing a titration, diagnosing an equipment malfunction, or landing the space shuttle. Many physical simulations are also procedural simulations, for not only is the physical entity imitated, but also the student’s performance must imitate actual procedures of operating or manipulating it. In fact, the primary focus of a simulation is usually procedural, and the simulation of various physical objects is therefore necessary to meet the procedural requirements” (Alessi and Trollip 1985).

Computer-based simulations portray complicated, difficult-to-obtain or expensive machines or pieces of equipment. Highlighting or showing only the necessary controls and gauges helps to eliminate distracting, excessive information and allows the student to concentrate

on the lesson at hand. Levels of complexity (often the addition of more sources of information) can be increased as the student advances in understanding.

4.4 Games

“Games may or may not simulate reality, but they are nearly always characterized by providing the student with interesting challenges” (Alessi and Trollip 1985). It is difficult to define, exactly, what an instructional game is as there are many types that lend themselves to different methods of learning. Occasionally, games created for entertainment do a good job of teaching. One familiar example is the board game *Monopoly*, which motivates thinking about finance and real estate. Logic-oriented, problem-solving games, psychomotor games (combining intellectual and mechanical abilities), and even role-playing games could have useful applications in graphic arts training. An interesting and complex role-playing game created for Harvard Law School by the American Video Institute at Rochester Institute of Technology focuses on lawyer/client interaction. This laser disk project is worth examining if this type of instructional technology is of interest.

Games use entertaining and/or fantasy-oriented scenarios to motivate play, interaction, and thus, learning. “Games provide a safe way of acting out a more dangerous reality, such as in war games, computerized contact sports, or investment [or estimating] strategies. This in turn encourages the players to explore alternative approaches in the game with the knowledge that failure at worst means losing the game. There are no real consequences” (Alessi and Trollip 1985). Certainly, many will remember Gulf War pilots recalling precise strikes against Iraqi installations as being just like the games they played in computerized training.

5.0 Producing computer-based training

Years ago, science fiction and fairy tale lovers might have watched a television program on the making of the movie *Star Wars*. Viewers were awed by the level of technical complexity and hours of tedious detail work needed. Artists, costume designers, writers, engineers, animators, talented special effects people, computer programmers and even pyrotechnicians were all part of the production. When one seriously considers making computer-based training that incorporates today's multimedia tools, all of the previously mentioned roles (save that of the pyrotechnician) also need to be addressed. Few people are a one-man band when it comes to producing these lessons. Many hours and resources are used to create a few minutes of interactive, multimedia courseware. As mentioned earlier, Xerox poured nearly 1,300 hours of a team's efforts into one quality hour of computer-based training. This production was on the high-end of the quality and content scale, however.

One should beware of alluring marketing packages from software and hardware vendors. Many do a great job of convincing people that they shouldn't be paying others for what can be done in house—with their products. Although it may be true that many projects can be done in-house, not many potential buyers have the experience in producing computer-based courseware nor do they have sufficient background in fundamental components they might like to use—animation, for example. Either deficiency means spending extra time to learn new programs and trying unfamiliar techniques. Software and hardware vendors do, however, usually offer some type of user support for those who need assistance in using their product. Sometimes help is even offered through on-line bulletin boards. Computer user groups in various cities are a good place to ask for assistance. Also, training programs do exist to help the inexperienced get involved with multimedia tools.

“The demand [for training] grows daily as people who believe they are getting ‘multimedia’ in one shrink-wrapped package realize that interactive presentations demand knowledge of many hardware and software tools plus expertise in video, audio, graphics, and animation” (Waltz 1991). *Publish* provides a helpful directory of instructional contacts which is provided as Appendix I.

5.1 A team effort

Once future training developers know what is involved in producing computer training materials, it becomes easier to systematically organize, assemble resources (human and material), and get the job done. Projects are almost always a team effort. The project manager must convey a vision of the finished learning experience to other team members which might include the following: an instructional designer, an interactive media designer, a graphic designer, video producers (if needed), programmers, artists, and animators. Although there may not be an individual to specialize in each of these areas, these roles constitute a functional team.

The project manager “creates the training situation in mind, visualizes the talent and props to be used, hears the words being spoken, sees the view angles and knows how the trainee will interact with the screen. If a computertext screen is to be used, then [the project manager] determines the words, the spacing, the rhythm of the presentation. As choices are developed, branching specifications are determined as well as the form of remediation to be used” (Smith 1987).

An overall plan, or blueprint, must be made perfectly clear to all team members. Mental visions must be made concrete in order to be communicated, edited, or even eliminated. “The mechanism for doing this is a preprinted form, which we have called a script page and others a Super StoryBoard, on which are provided ways to describe the mind’s eye as it might appear to the delivery system. The script page is designed to reflect all the capabilities of the delivery system, and therefore the page form will be different for each system. A particular script page design will certainly include areas for description of visuals and dialogue. It includes page identification and a template for branching specification. It must include a place for specification of delivery mode (computer, videodisk, graphics, etc.) and any other system characteristics which will be used by the training. If the software system calls for data in a particular way, then the script page should call for specification in a similar way” (Smith 1987).

5.2 Eight steps to follow

Alessi and Trollip break the process of developing computer-based instructional materials into eight steps: “1) Define your purpose; 2) Collect resource materials; 3) Generate ideas

for the lesson; 4) Organize your ideas for the lesson; 5) Produce lesson displays on paper; 6) Flowchart the lesson; 7) Program [or produce] the lesson; 8) Evaluate the quality and effectiveness of the lesson” (Alessi and Trollip 1985). Although some developers may dispute the order of certain items in this list such as whether idea generation should come before collection of resource materials or whether some steps will have to be repeated, the basic process remains the same.

Before embarking on the process of assembling a lesson, make sure that student objectives will be met—”determining what you want your students to know or be able to do at the end of your lesson. Many instructional designers recommend a more complete elaboration of very specific learning objectives for both the end of the lesson (terminal objectives) and intermediate points during the lesson (intermediate or enabling objectives)” (Alessi and Trollip 1985).

5.3 Harvard Law School example

In the American Video Institute’s training program for Harvard Law School, a video laser disk was used to store images and sound. This choice resulted from a careful investigation of what hardware would be used by Harvard—a standard IBM personal computer and a television. Without the storage capacity of huge gigabyte hard drives, the laser disk served as a fast, high capacity medium. The creation of such a disc has its own advantages and disadvantages, and it does dictate how production of courseware proceeds. Nevertheless, certain phases remain constant in any such project.

Using a storyboard and a tree-like diagram that shows the branching possibilities within a training sequence, one may get a clearer picture of where trainees may find themselves after making choices and following paths. At each decision and stopping point, there must be new information. This is where the labor of producing computer-based training manifests itself. Images, sounds, text, and graphics must be produced for each of these instances. In the case of the Harvard Law School laser disc, photographs of characters and case “evidence” as well as audio portions had to be recorded—on the scale of thousands. Also needed were computer screen graphics (the user interface) and text. These raw elements are common to many interactive computer projects and require a certain level of skill to produce in an aesthetically pleasing way. This is also where help with illustration, 3-D, animation, and video/audio capture comes into play if the production team lacks these abilities.

For laser disk production, images and sound complete the first phase of production in respective audio tape and video tape forms. In the second phase, all is merged to video tape, including screen graphics and text. A pre-master laser disc is then made. Pioneer sells the technology that makes this possible for in-house use. Meanwhile, programmers work on the scripting involved to make the whole system work. Finally a laser disc master is produced and the finished software lets the training proceed smoothly.

If a laser disc is not used, one must consider average access time of an extremely large hard drive and deal with the memory size requirements of images and sounds to be incorporated. For example, 12 minutes of Apple's Quicktime video occupies about 43 megabytes on a hard disk drive. Still video and sound vary depending on size of the picture, duration of the sound, etc. There are also data compression schemes to consider that save on memory, and this introduces a loss of quality in the overall image projected to the user. Those creating the computer-based training materials must decide how to compromise speed and memory requirements versus the "look and feel" of the final product. Again, average production time for all of the above is 400 hours per one hour of finished product. Plan accordingly.

A systematic approach to developing computer-based training materials will contribute to the effectiveness of the finished product. When a team gets sloppy, unanswered questions and small, dysfunctional portions of a lesson that are uncovered only serve to distract the student and hurt the perceived credibility of the courseware, its contents, and its creators. "Conditions in industrial and military settings are ideally suited to employing systematic methods. Learning outcomes are usually clearly defined since they are based on specific job descriptions or known management needs. These are clarified even further through job analysis and task analysis. In addition, these organizations are able and willing to make the substantial resource investment necessary to employ in-depth instructional design methods, especially since the costs can be justified by increased worker productivity and amortized over a large population of potential users" (Roblyer 1988).

5.4 Preliminary evaluation

Before releasing computer-based training materials to their intended audience, formative evaluation should have been implemented. If there is sufficient time, earlier evaluations should be scheduled to highlight troublesome points before too much effort has been

expended. In these evaluations, a sample group of people, having no affiliation or tie to the project takes the interactive training for a test drive.

“Materials are tested with students or classes of students from the target population. Data should be collected on student performance, student attitudes toward instruction, and teacher attitudes. At least one member of the design team should watch students as they go through the instruction to determine any problems they may have with computer use or with understanding what they should do. Students should be interviewed to get their reactions to the materials’ ease of use and interest. It is essential that the actual screens be proofread to eliminate any misspellings, incorrect grammar, or typographical errors. Based on this research, the team will revise the materials as necessary and finalize them” (Roblyer 1988).

Besides course content and general computer use, careful computer-based training developers will also pay close attention the interface design. An effective interface not only looks appealing, it ties the courseware together consistently, aids navigation through the material, and provides help along the way. Apple Computer cautions interface designers to think especially hard about the following ten considerations: Use of metaphors, direct manipulation of objects on the screen, “see and point”-ability, consistency, WYSIWYG display, user control throughout, feedback and dialog via system beeps and text information boxes, forgiveness of common user slips, perceived stability, and aesthetic integrity. The HyperCard stack checklist is a good guide in formulating an interface. It may yield considerations and suggestions that are not obvious to the new courseware developer (see appendix II).

6.0 Software & hardware requirements

“Computer-based training” and “interactive media” do not necessarily mean “multimedia.” But today, it is getting more and more difficult to ignore tools like sound, animation, color, and even video in seemingly anything delivered by a computer. Although there is much marketing glitz associated with multimedia capabilities, there are real and practical applications that can only enhance, or even make possible, computer-driven instruction. Most courseware is not produced merely to entertain, but it cannot hurt to deliver technologically up-to-date materials to a channel-surfing audience.

This year’s computers are delivering multimedia capability to the masses. Apple Computer has been leading the way by incorporating sound and a relatively fast graphic display engine—Quickdraw. Now Apple has released an easily-installed software file for each user’s system that enables the computer to play back video—Quicktime. Meanwhile, in the DOS world, Microsoft recently released its multimedia extensions. That and the Multimedia Personal Computer (MPC) standard has developers everywhere rushing to create new products. MPC sets a common configuration for multimedia hardware and software that includes digital audio capabilities, a CD ROM drive, and Microsoft’s Multimedia Extensions for Windows 3.0. Soon exposure to multimedia capabilities will be widespread. Many will even come to expect certain finesse from professionals using these tools, which puts pressure on those developing any type of computer instruction.

6.1 Authoring software

Luckily there are established authoring products (software) available on both Mac and PC platforms that enable orchestration of user input and multimedia output with minimal developer angst. “In general these programs incorporate one or all of the following data types: text sound, animation, graphics, MIDI, and/or CD audio and video. To incorporate this data into some type of coherent presentation, the application must rely on specialized hardware (and a lot of technical know-how) to deliver the data” (Canning 1992). Not only will courseware developers need an authoring program, they will also use image editing and paint software such as Adobe Photoshop, illustration packages like Aldus Freehand, sound recording equipment, a MIDI interface, video source, animation, and possibly three-dimensional modeling. Try keeping a few of those going simultaneously on four megabytes of RAM.

Canning's article, cited above, on authoring tools is perhaps the most complete compendium to date on the subject. Anyone seriously interested in building anything that utilizes multimedia capabilities should read the whole 12-page article. Canning accurately points out that each program is best suited for a certain environment—from highly visual animations to hypertext to user-tracking and evaluating needs. *Infoworld* evaluated five packages: Authorware Professional for Windows 1.0 from Authorware Inc., IconAuthor 4.01 from AimTech Corp., MacroMind Director 3.0 (Macintosh) from MacroMind/Paracomp, Microsoft Multimedia Viewer 1.0 (MDK), and Toolbox & Media Resource Kit from Asymetrix Corp.

Infoworld discusses the respective programming environments for the most part. It is the heart of a computer-based training application. A high quality environment requires “basic flow-control operations such as looping and conditional branching, along with calculation and data manipulation capabilities (utilizing the equivalent of variables, Boolean logic, and simple mathematic calculation). Debugging facilities, string manipulation tools, timing controls, dynamic file I/O, and the capability to compile the finished application are added, useful features” (Canning 1992).

The software component of a training system must be able to deal with a variety of other media inputs and their respective file formats. Without these abilities, trainers will be shortchanged in the tools they should be able to use well into the 90s. *Infoworld* required the programs analyzed to use static and time-based media like video, sound, or animation. “For a satisfactory score, we require the capability to import bit-mapped files in two or more file formats common to the tool's native environment. These may include BMP, PCX, TIF, GIF, TGA, or DIB for Windows and TIFF or MacPaint for the Macintosh. We also require ASCII text import from disk or clipboard and playback of sampled sound and CD-Audio” (Canning 1992).

6.2 Macintosh-based software

On the Macintosh platform, the choice is easy. Canning presents the only multimedia authoring tool, MacroMind Director 3.0. Hypercard 2.0, supplied with every Apple Macintosh computer, serves as a good prototyping program. With some help from external commands written in Pascal or C, it can display color pictures, interface a laser disc player, and start other applications. It cannot, however, display color screens, do animation or

compile projects produced to operate without the application as can Director. MacroMind Director's scripting language, "Lingo," is touted as being "similar to HyperCard's scripting language." This is either a plus or a minus depending on your viewpoint and experience. For many graduate students this year, Lingo was similar enough to make commands look and sound familiar but different enough to make writing scripts annoying. It takes a little assimilation. Director "movies," as its files are called, can be run on a PC with a software player; they must be developed, however, on a Mac until a DOS version is available. Overall, MacroMind Director 3.0 is especially strong in animation and has no real weaknesses in other categories such as user support or its programming environment.

6.3 DOS-based software

Jumping to the high end of the DOS environment, Authorware Professional for Windows represents perhaps the best program for monitoring user interaction. It is "a good choice for training systems and educational organizations" (Canning 1992). Animation and general organization of work in progress via icons on a flowchart are heralded as "elegant." This authoring program is touted as being extremely easy to use. It's too bad user-friendliness and wallet-friendliness don't go hand-in-hand. Authorware lists for \$8,000. If that's not enough, run-time licensing fees and technical support cost \$1,200 annually. This means any training application made on Authorware and then distributed for profit without payment of the licensing fee is illegal. These licensing fees are one more figure to consider when summing up costs associated with bringing this type of production in-house. However, not all software companies have these licenses. Microsoft's Multimedia Viewer, for example, produces a compiled, executable file that is free for distribution.

Infoworld's lab regarded IconAuthor by AimTech to be a top-quality program and a good value. It is a "full-feature development product with utilities for developing source materials for multimedia titles. Due to its high price, \$4,995, IconAuthor is geared toward large institutions with a variety of applications to build, especially training systems. Its flow-chart programming paradigm allows it to be used by non programmers, although some training will be required. Its excellent tutorials and included vendor training should make it easy to adopt" (Canning 1992).

The most favorable list price for PC users (\$495) comes with Microsoft's Multimedia Viewer. One should not run out and buy it for their in-house computer production staff and

expect them to make training modules on the level of Authorware or even MacroMind Director. Multimedia Viewer cannot compare as its strengths lie elsewhere. This program is not for those who want to do animation and/or scripting of interactive projects because it lacks the tools. "It is best suited for building on-line documentation systems or educational material that involves lengthy descriptions" (Canning 1992). Hypertext-style linking is this program's forte. Consider this program for technical service representatives who are constantly on the road and need to download the latest solutions or system updates via their laptops.

As seen above, there are different software solutions for different needs. There are several more packages covered at length in *Infoworld*. The key is to carefully assess what type of training needs to be produced for new scanner operators, press operators, printing company customer service representatives and even MBAs new to the industry who need to know the processes involved in color reproduction. Only then can a software decision be made, from a program strong in animation to hypertext to tracking student progress and automatically adjusting the level of difficulty.

6.4 Hardware requirements

Hardware requirements represent an entirely different can of worms that, fortunately, is writhing less these days thanks to the computer industry's consolidation of approaches. MPC represents minimum standards of performance and capability in the PC world. Not only can a developer count on the MPC computer user having at least a 286 processor and a Super VGA monitor, a CD-ROM player can also be expected. The Macintosh environment actually benefits from its one-company origin in that all computers as of 1991 adhered to certain multimedia-ready standards. Sound, for example, was included in every computer whether the user wanted it in a Mac II si or not. These lowest common denominators represent a starting point for developers of training materials that will reach a wide audience with varying degrees of computer power.

Executing files and playing back multimedia possibilities is one thing. Creating them is another. Today's multimedia-ready computers in the PC market have the performance capabilities of IBM's Ultimedia, Compuadd's Model 333, or the Tandy 4033LX. Each of these machines has at least a 386-level processor running at 33MHz. Speed of the processor is important when playback of sound and animation are involved. Typical VCRs

play back about 30 frames per second; that is the speed that tricks the eye into seeing continuous motion. The faster a computer is able to display images, the better certain training types, like simulations, will look. Each of the above systems also includes at least a 105 MB hard drive and a CD-ROM drive. Two of the systems featured one megabyte of video memory to speed display redraw. IBM's Ultimedia "is capable of delivering 256 colors at 1,024-by-768 pixel resolution or 65,000 colors at 640-by-480 pixel resolution" (Copeland, McCarthy, and Kvitka 1992). IBM's machine is considered near or on top of the heap when it comes to graphics ability in the DOS world.

For audio, DOS-based computers rely on sound boards and mini stereo speakers. The Tandy 4033LX uses Creative Labs' Sound Blaster Pro sound card installed in an expansion slot. The IBM Ultimedia's sound capabilities are internally contained, similar to a Macintosh's. "An M-Audio adapter connects to a built-in high fidelity speaker with volume control" (Copeland, McCarthy, and Kvitka 1992).

6.5 Pre-configured solutions

One consideration worth mentioning is that some manufacturers are pre-installing multimedia computer components, software and all. For those who would rather not rummage around inside their new \$4,500 machine, it's worth the extra cost. "This may sound trivial, but if you've ever tried to install a multimedia setup from scratch and had to guess about which drivers get loaded when, you will appreciate pre-installed multimedia software" (Copeland, McCarthy, and Kvitka 1992). So if the thought of configuring systems makes computer-based training unthinkable, there are affordable solutions.

On the Macintosh, comparable systems can be built on everything from the Mac II si to the new Quadra 900. Apple's new Quicktime system extension will allow all computers to display compressed video incorporated into electronic documents whether they have a special video board or not. Quicktime sacrifices video display quality for sound integrity in its compression algorithm, but the technology represents an admirable start. If one wishes to actually record information to video on a Macintosh, a display board such as SuperMac's Video Spigot must be used. Quality of the video presentation does fluctuate slightly depending on the display board used during signal capture.

Multimedia features enable computer users to do what only proprietary systems could a decade ago. If this type of undertaking is to be completed in house, lots of hard disk storage (including removable cartridge drives) and RAM are needed to complete the job without undo frustration. No matter how manufacturers package multimedia systems, developers are going to need lots of power to create good courseware. On the Macintosh, for example, one would not want to start working on MacroMind, incorporating paint tools, color, etc. without at least eight megabytes of RAM. Of course, programs like Adobe Photoshop, Microsoft Excel, and Aldus Freehand might also be running to allow cutting and pasting of graphic information and photos into the production. What will your system need to keep everything running for the team that builds computer-based training? Needs will vary, but they can be estimated by adding system memory (RAM) requirements for the operating system, the authoring tools, and other programs to be used simultaneously.

7.0 Empowered, but enabled?

By now many prospective computer-based training developers have either a better understanding of what this new frontier entails or they've found a company to do it for them. Actually, a few exist. Carousel Mediaworks in Rochester, NY; Ikon Interactive Multimedia in San Francisco, CA; and Pixel Ink Consultants, Inc. of San Francisco are companies worth contacting for starters.

Never before has the quality of software and hardware that is available today for meaningful interactive teaching and presentation been in the hands of the masses. Computer-based training has previously been the domain of developers who created proprietary material for companies and institutions that could afford it. Now that "non programmers" are using authoring systems to direct interactivity and multimedia output, eyes are opening to the possibilities that abound in the education, business training and entertainment realms.

One can buy the hardware and the software to create computer-based training modules in house. All that need be supplied is the lesson plan and course content. What might not exist in every company is the level of expertise needed to make systems that really teach, animations that work and look like they are supposed to, hardware/software combinations that are stable performers, and the time needed to perform formative evaluations on courseware before it is sent to its intended audience.

Interactive media applications are here to stay, and industry only stands to benefit from the technology. Conventional classroom instruction can be supplemented by or partially replaced by computer-based training. The technology has a proven track record when it comes to student achievement, savings in time, and positive student attitude. At a time when quality products and a knowledgeable work force are recognized as fundamental to the United States' future in the global marketplace, industry cannot afford to pass up cost-effective re-education of its employees.

8.0 References & sources cited

Alessi, Stephen M. and Stanley R. Trollip. *Computer-Based Instruction: Methods and Development*. Englewood Cliffs, NJ: Prentice Hall, 1985.

Apple Computer, Inc. *HyperCard Stack Design Guidelines*.
Reading, MA: Addison Wesley, 1989.

Arajas, Bridget Kinsella. "The Discussion Continues...and Continues," *Graphic Arts Monthly*, April 1991, p.100.

Bork, Alfred. "Interaction: Lessons from Computer-based Learning," *Interactive Media: Working Methods and Practical Applications*. Ed. Diana Laurillard. Chichester, England: Ellis Horwood Limited, 1987; pp. 28-42.

Canning, Jim. "Multimedia Authoring Tools," *Infoworld*, March 9, 1992, pp. 76-93.

Copeland, Ron , Michael E. McCarthy and Andre Kvitka. "Bring Your Flashy Applications to Life: Four Approaches to Multimedia Hardware," *Infoworld*, March 9, 1992, p. 96.

Cohen, Howard and Steven Moonitz. "Desktop Training: Getting the Right Mix," *Folio*, May 1991, pp. 74-6.

Ferris, Fred. "Basic Training," *American Printer*, May 1991, pp. 24-7.

Frenkel, Karen A. "Peeking Behind the Interface," *Publish*, July 1991, pp. 58-68.

Humphries, Charles E. "Who will Train In-Plant Staff?," *In-Plant Reproductions*, May 1991, pp. 28-9.

- Kulik, James A., Chen-Lin C. Kulik , and Peter Cohen. "Effectiveness of Computer-based College Teaching: A Meta-analysis of Findings," *Review of Educational Research*, 50: 525-544, 1980.
- Kulik, James A., Chen-Lin C. Kulik , and Peter Cohen. "Instructional Technology and College Teaching," *Teaching of Psychology*, 7: 199-205, 1980.
- Kulik, James A., Robert L. Bangert, and George W. Williams. "Effects of Computer-based Teaching on Secondary School Students," *Journal of Educational Psychology*, 75: 19-26, 1983.
- Mitchell, Christine and T. Govindaraj. "Design and Effectiveness of Intelligent Tutors for Operators of Complex Dynamic Systems: A Tutor Implementation for Satellite System Operators," Georgia Institute of Technology, *Interactive Learning Environments*, 1:3, 1991; pp. 193-229.
- Montague, William E. "Promoting Cognitive Processing and Learning by Designing the Learning Environment," *Instructional Designs for Microcomputer Courseware*. Ed. David H. Jonassen. Hillsdale, NJ: Lawrence Erlbaum Associates, 1988; pp.125-150.
- Roblyer, M.D. "Fundamental Problems and Principals of Designing Effective Courseware," *Instructional Designs for Microcomputer Courseware*. Ed. David H. Jonassen. Hillsdale, NJ: Lawrence Erlbaum Associates, 1988; pp. 7-32.
- Smith, Richard C. "The Creation of an Integrated IVD Training Curriculum," *Interactive Media: Working Methods and Practical Applications*. Ed. Diana Laurillard. Chichester, England: Ellis Horwood Limited, 1987; pp. 132-42.
- Waltz, Mitzi. "Back to School," *Publish*, November 1991, pp. 88-93.

Appendix I: Multimedia training resource guide

*The following is a list of sources for multimedia
or training referrals:*

Amiga East Bay Users Group
(415) 845-4814

**Art Center College of Design
Computer Graphics Department**
1700 Lida St.
P.O. Box 7197
Pasadena, CA 91109-7197
(818) 584-5000

Awest (an Amiga user group)
San Francisco, CA
(415) 584-8406

**California State University-Hayward
Art Department**
Hayward, CA 94542
(415) 881-3111

Center for Electronic Art
329 Bryant St., #3D
San Francisco, CA 94107
(415) 882-7063

Electronic Directions
220 E. 23rd St., #503
New York, NY 10010
(212) 213-6500

IBM
4111 Northside Pkwy.
Atlanta, GA 30327
(404) 238-4526

Imagine
32 North Washington, #14
Ypsilanti, MI 48197
(313) 487-7117

Knowledge Industry Publications
701 Westchester Ave.
White Plains, NY 10604
(914) 328-9093

**Lehigh University
Educational Technology Center**
Bethlehem, PA 18015
(215) 758-3231

MacroMind
600 Townsend St., #310
San Francisco, CA 94103
(415) 442-0200

**Massachusetts Institute of Technology
Dept. of Media Arts and Sciences**
Room E15-224
Cambridge, MA 02139
(617) 253-5114

Mind Over Macintosh
2902 B Colorado Ave.

Santa Monica, CA 90404
(213) 829-2756

New York Mac User Group
(212) 986-0852

Quantum Computers
5003 Tacoma Mall Blvd.
Tacoma, WA 98409
(206) 475-7000

Quickstart Computer Training
5862 Bolsa Ave., #103
Huntington Beach, CA 92649
(714) 894-1448

San Diego Mac User Group
(619) 270-1936

Solutions!
4526 Fountain Dr.
St. James Village
Wilmington, NC 28403
(919) 395-0233

Triple A Amiga User Group
28 Alan Way
Martinez, CA 94553
(510) 935-0829

Graphics Plus
3760 South Robertson
Culver City, CA 90232
(213) 559-3732

Publish, November 1991, p. 90

Appendix II: Interface guidelines

Selections from the Stack Design Checklist in

HyperCard Stack Design Guidelines, Apple Computer, Inc.,

Addison Wessley, Reading, MA ,1989 ,pp. 194-204.

1. Who's your user?
2. Previous experience?
 - Computers
 - Macintosh
 - IBM compatible
 - Subject matter expertise
3. Context in which user will be working with application.
 - Alone
 - With other software
 - With what hardware
 - Over a network
 - As part of a training class
4. Projected machine environment
 - versions of authoring system
 - user level
 - computer models and memory
 - printers
5. Special markets?
 - International considerations
 - Which countries
 - Disability audiences (and for which disabilities)

Subject matter

6. What's your subject matter?
 - general purpose
 - how much will the application cover
 - what won't it cover
7. How much memory have you to work with?

- 800K disk
 - 800K disks
 - 1.4 MB disk(s)
 - 44 MB (Syquest® removable)
 - CD-ROM disk (656 MB)
8. What resources do you have?
 - time
 - developers, graphic designers, sound composers, other
 - money
 - machines & equipment
 9. What is the specific purpose of your project, given the subject matter's scope, size, space limitations, and your development resources of time, people, and money?
 10. How is it most appropriate to present subject matter to users?
 11. What natural sections does the subject matter divide into?
 12. What functions must it perform?
 13. What is the tone?
 14. What presentation method seems most suitable?
 - slide show
 - demo, with continuous animation
 - training/structure & guidance
 - game
 - tool or utility
 15. Will the project be identifiable as, say, HyperCard? Will the look and feel resemble the Macintosh or the DOS or the MS Windows interface? Will it have a completely unique look?
 16. Is there a real-world metaphor to support the project?

Navigation

17. Is navigation within the application easy?
18. Are navigation tools placed in exactly the same place on the screen throughout?
19. Have you provided the user with information about the structure of your project?
 - maps
 - diagrams
 - menus

- a way to tell where users have been
20. Have you chosen an appropriate structure?
21. Does the project include the following elements?
- introduction or opening
 - help functions
 - something for the user to do right away
 - a home base or reference point
22. When you conduct a preliminary test, do people use it correctly?
- or do they ignore the help function
 - mutter in frustration
 - comment positively
 - use functions
 - go through project smoothly

Graphics

23. What is your visual look?
- art-deco
 - businesslike
 - medieval, etc.
24. How have you planned screen layout?
- underlying grid pattern
 - edges of grid used for permanent navigation functions
25. Which of the following will you need to design?
- buttons and icons
 - fields for text
 - backgrounds
 - illustrations
 - scanned images
 - animation sequences
 - screen titles
26. Have you planned for typography?
- font styles that visually match the tone of your stack
 - fonts big and bold enough to be easily read
 - text laid out on a grid

- line width and spacing that make text readable

Writing

27. How polished and appropriate is your writing?

- concise expression
- works with graphics to convey meaning
- works with sound to convey meaning
- flows well from screen to screen; sounds good if read aloud
- is sufficiently independent on each screen so that users understand meaning
- has the right tone for your audience
- has been checked for accuracy and editorial correctness

Sound

28. Do you tell the user how to deal with sound?

- instructions for turning down sound
- instructions for eliminating sound
- instructions for installation onto hard disk if required for sound

29. Are sounds associated with the project's operation in a consistent way?

- visual effects
- major navigational transitions
- project content
- user feedback
- progress indicators
- a running sound track

30. Have you tested sounds completely?

- on all intended machines will all possible configurations of RAM and disk space
- with and without a hard disk
- at different volume levels
- with several repetitions, to make sure they're not annoying

Testing

31. Test several times during the design and development of project

32. Don't instruct testers how to do whatever you're testing

33. Set an objective

34. Decide how to record user trials
35. Determine the setting (microphones, videotape, or no distractions)
36. Find users representative of target audience
37. Use the results

Appendix III: Creating “Platemaking Fundamentals”

Sources of information:

Prof. Mark Collien, American Video Institute, Rochester Institute of Technology

Prof. Frank Cost, SPMS, Rochester Institute of Technology

Prof. Cliff Frazier, SPMS, Rochester Institute of Technology

Prof. Steve Kurtz, Computer Science, Rochester Institute of Technology

Prof. Len Leger, SPMS, Rochester Institute of Technology

Mr. Jim Manning, T&E Center, Rochester Institute of Technology

Mr. Franz Sigg, T&E Center, Rochester Institute of Technology

Brunner Switzerland. *Directions for Use: Ugra Plate Control Wedge 1982*. February 1987 edition.

Eastman Kodak Company. *Kodak Litho Plates: Technical Data Pub. Q-231*. Rochester, NY; 1990.

Goodman, Danny. *The Complete HyperCard 2.0 Handbook*, third edition. New York: Bantam Books, 1990.

Hartsuch, Paul J., Ph.D. *Chemistry for the Graphic Arts*. Pittsburgh, PA: Graphic Arts Technical Foundation, 1979.

Reed, Robert F. *Offset Lithographic Platemaking*. Pittsburgh, PA: Graphic Arts Technical Foundation, 1967.

Young, L.C. *Materials in Printing Processes*. New York, NY: Hastings House Publishers, 1973.

Hardware used:

Apple Macintosh II ci; 8 MB RAM/80 MB hard disk, 8/24-bit color, RAM cache

Syquest removable hard drive—45 MB

Canon Xapshot camera, disks, and video conversion hardware and software

Agfa focus flatbed scanner

La Cie Silverscanner (flatbed)

Software incorporated:

Hypercard 2.1 + various external commands

ResEdit 2.1.1

ResCopy 2.0

Adobe Photoshop 2.0

McView II (with Agfa scanner)

MacroMind Director 3.0 (animations)

MacroMind Player 3.0

Compact Pro 1.3

Time involved:

Approximately 100-120 hours

Average amount of work per hour of interactive learning: 400-480 hours

Time-consuming tasks: searching for input (usually images) that does not exist; drawing cards/working around authoring program's pictorial deficiencies, like reverse type; programming and debugging.