Imagine that you are selecting the delivery method for a soon-to-be-developed training course. Although most of the factors favor computer-based training (CBT), your estimates show that development costs will exceed budget limits if you custom design a CBT course. Yet, when you review packaged CBT products, none provide a perfect fit with your training goals and objectives.

This is the challenge Arthur Andersen & Co.'s professional education division faced when we wanted to bridge the gap between a packaged CBT program and our desired course goals and objectives. But we came up with a unique approach to make it work.

Cost-Effective Computer-Based Training

By Robert J. Campbell

The challenge

The course in question was our fundamentals of accounting course, required for nonbusiness majors in our management information consulting division. Originally this program was in a textbook-based, self-study format with supporting activities and supplemental content in paperback guides. Because of the firm's policy of continually upgrading the quality of training, the course was scheduled to undergo significant revisions.

Based on our evaluation of potential instructional approaches, we identified CBT as the most appropriate medium for the revised course, and we found a packaged CBT accounting tutorial that came close to meeting our needs. The tutorial consisted of 10 modules, each

Campbell is an instructional designer at Arthur Andersen & Co. in St. Charles, Illinois.

taking 30 to 60 minutes to complete and addressing as many as eight separate concepts.

From the start, however, it was apparent that this stand-alone tutorial wasn't complete enough to meet all our training goals and objectives. It appeared that we would either have to custom develop the CBT at high costs, or use the packaged accounting tutorial as stand-alone training and compromise our training goals and objectives. Since neither of these choices was acceptable, we needed an alternative. After much thought we decided to fill the gap between the content covered in the packaged accounting tutorial and our original course specifications by having participants complete paper-based materials following each module.

Even this solution posed several problems. Waiting until the end of a module to reinforce *all* the concepts within that module meant delaying appropriate reinforcement of each concept. In addition, we would be presenting concepts in between the presentation of a main concept and its supporting activities. Also, if we wished to cover additional content we would be forced to present it at the end of a module rather than where it should logically fit. It wasn't hard to imagine how confused and frustrated the participants would be if we chose this approach.

We needed to find a way to branch out participants to the paper-based material relevant to a concept right after we covered that concept in the tutorial. Then we could integrate the two mediums effectively.

The solution

Concurrent training seemed to be the answer to our challenge. This kind of training means using CBT to teach computer software while that software is running. An example of concurrent training is a computer-based tutorial that teaches spreadsheet software: Participants can receive information about the spreadsheet software while that software is running by using training "windows" that appear at key locations. The concurrent training program can also use examples and activities that directly interact with the spreadsheet software. The advantage of concurrent training is that participants directly interact with the software application being learned.

How did concurrent training help to solve our problem? We reasoned that if we could use concurrent software to develop on-line training for other software applications, we probably could use it to *modify* stand-alone CBT. We could take the accounting tutorial and treat it as our software application. Using concurrent software we could then place windows directing participants to the supporting paper-based materials at key points within each module of the tutorial. This would provide participants with immediate reinforcement of each concept. It would also allow us to introduce new material via the paper-based medium wherever we felt it appropriate.

There were other advantages to this approach. First, the savings over custom development would be tremendous. Also, we would be able to meet *all* our training goals and objectives. And if this approach were successful, it would become cost effective to use CBT in lower priority projects for which CBT would be the most effective medium.

The process

The approach outlined below is a logical extension of the traditional use of concurrent training. Without a specific model to follow in designing and developing our course, we drew upon existing instructional design models as we refined our approach to the project.

Organization

Our project team consisted of two full-time and four part-time members. The full-time members were a content expert-computer programmer and an instructional designer, who carried the primary responsibility for the design and development of the course. The four part-time members included a line manager responsible for content, two education managers, and a senior instructional designer. These four members were primarily responsible for planning the work and conducting quality assurance reviews throughout the project. Figure 1 outlines the project tasks by specific skill type.

Course specifications

Because the new course was to be a revised version of an existing course, many of the course specifications were already established. But there were some content issues we needed to resolve. For instance: Should the learn-ing objectives, content scope, and level of detail of the existing course remain the same for the new course? If not, what changes should be made?

Because of these issues and others like them, we conducted a course needs analysis, which ensured that we were developing a product that was instructionally sound as well as met the specific needs of our participants. The final course specifications came from the results of this analysis.

Existing materials evaluation

At this time we also needed to evaluate existing materials developed internally as well as those available from vendors. Even though we had identi-

---

Figure 1 - Project tasks by specific skill type

Position	Degree of Involvement	Responsibilities
Content or line manager	Part time*	• high-level project planning • course specifications review • final review
Unit or senior education manager	Part time	• high-level project planning • final review
Education manager	Part time	• project planning at both high and detailed levels • course specifications review • review throughout the project • software negotiations
Senior instructional designer	Part time	• project planning at both high and detailed levels • course specifications review • review throughout the project • supervision of pilot test
Instructional designer	Full time	• detailed-level project planning • development of course specifications • existing materials evaluation • concurrent software review and selection • detailed product design • primary responsibilities for storyboard development • secondary responsibilities for programming • materials development • pilot-test organization and conduct • postpilot-test revisions • project documentation • project time control • final review
Content expert/computer programmer	Full time	• detailed-level project planning • development of course specifications • concurrent software review and selection • detailed product design • secondary responsibilities for storyboard development • primary responsibilities for programming • materials development • pilot-test conduct • postpilot-test revisions • final review

Part time refers to less than two hours a week

fied an accounting tutorial that we thought would work in the new course, we wanted to confirm that there were no other CBT packages that better suited our needs. After contacting a number of vendors and reviewing materials developed internally, we were unable to locate a more appropriate CBT accounting package than the first one we had identified.

Although we had tested the accounting tutorial primarily for content, we needed to evaluate its instructional integrity before using it in our course. Here are some of the questions we asked during this evaluation:

■ Are the training objectives in the course specifications covered, and, if so, are they covered at an appropriate level of detail?
■ Is the information presented in a logical, organized manner?
■ Do screens reflect the use of sound design principles? Specifically: Are screen standards apparent, and, if so, are they consistent? Are color, graphics, and text used appropriately?
■ Is the product user friendly?
■ Is the product truly interactive, or does it simply transfer text from a paper-based medium to a computer-based medium?
■ What would we need to do to modify the product so that it would meet all of the course specifications?

Overall, our assessment of the tutorial was favorable. By including supplemental paper-based materials, the tutorial would work well as the body of our new course. Our next step was to construct a high-level design of the course.

High-level course design

At this point we looked closely at discrepancies between the packaged accounting tutorial and our course specifications. The primary weakness of the accounting tutorial was that it did not contain enough practice of the principles taught to adequately satisfy our training requirements. It also lacked coverage in what we believed were key content areas, and it treated some material as optional that we considered mandatory.

Next we determined how we could structure the course to eliminate these discrepancies. We believed the course should be driven by a paper-based guide, and we decided we would divide the course into five segments, each of which we would further divide

into topics that correspond to modules in the accounting tutorial. Each topic in the guide would begin with an introduction, statement of topic objectives, and instructions directing participants to begin a module of the tutorial.

Then we would use concurrent software to overlay windows at key locations in the tutorial. These windows would contain branching statements directing participants to either complete activities or read additional content found in the guide. This would result in weaving together the computer-based accounting tutorial and paper-based guide by simple branching statements. It would also ensure that participants would be able to complete activities and additional content at the appropriate times, rather than have to wait until the end of a module.

We reasoned that if we could use concurrent software to develop on-line training for other software applications, we probably could use it to *modify* stand-alone CBT

We also decided to use the concurrent software to lock participants into following a predetermined sequence in an area where the accounting tutorial provided optional coverage of some content. This ensured that participants would cover material presented in the tutorial that we believed was too important to be treated as optional.

Once these basic criteria were established for the overall design of the course, our next step was to evaluate concurrent software.

Concurrent software evaluation

An important group in our professional education division is the research and evaluation services department. One of this group's responsibilities is to evaluate new educational technologies. Based on their research, we identified two viable concurrent software programs and evaluated these programs against criteria in three categories: flexibility, learning curve, and cost.

The first program was fairly easy to learn as it was completely menu driven. The cost was reasonable, but it soon became apparent that it didn't provide the flexibility our project required. It would work only if participants followed a predetermined path throughout the accounting tutorial *and* if they completed the tutorial at one sitting. Due to the nature of the tutorial, however, we needed a concurrent software program that would allow participants to access a screen containing a concurrent window by various means—whether by jumping ahead, by jumping back, or by turning the computer off and on and returning directly to the screen.

The second concurrent software program was also fairly inexpensive, but it had a substantially longer learning curve. This was because in addition to being menu driven, it also had a programming language. The combination of these features made this program extremely versatile. It also had a special screen-capture feature allowing reference to a screen through a string of text unique to a specific location on the screen. This feature provided the referencing capability we required for the concurrent windows.

Once we had decided on both the accounting tutorial and the concurrent software program, we were ready to begin negotiations with the respective vendors.

Software negotiations

Our intended use of the accounting tutorial presented an additional issue we had to resolve during our negotiations. Even though we didn't plan to change the actual program code of the accounting tutorial, by wrapping concurrent software around it we would, in effect, be changing the tutorial. We were straightforward with the accounting tutorial vendor in describing how we planned to customize it, and we concluded an agreement that gave us permission to carry out our plans.

The research and evaluation services department conducted negotiations with the vendor of the concurrent software program. We learned that in addition to purchasing the development software, we would incur a cost each time a participant used the training. We were offered two options: a one-time fee or a flat rate per participant. The amounts involved were negotiable, and we were able to reach an agreement that satisfied both parties.

Upon concluding negotiations with both vendors, we proceeded with the next step: detailed course design.

Detailed course design

The principle task here was to define the details of the high-level course design. We determined exactly what additional activities and content we needed and where we should place them. We identified format standards for both the paper-based materials and the concurrent windows that would modify the accounting tutorial. We also developed a "storyboard" for the windows that we planned to create with the concurrent software program. This storyboard included the location and design of each of the concurrent windows to be developed.

As opposed to conventional concurrent training, we determined that the concurrent windows that were to be added should be "seamless." Concurrent windows are usually set apart from the subject software by either color, shape, size, screen position, or a combination of these. This ensures that participants don't confuse the training with the subject software itself. Since we were customizing stand-alone CBT, our goal was to modify the accounting tutorial with concurrent software in a way that the participants wouldn't be able to distinguish one from the other.

After we completed the detailed course design, we were ready to develop the course.

Course development

The detailed course design created in the previous step represented the blueprint for the course. Development proceeded closely to plan. Much that we learned during course development was due to trial and error. We modified the programming plan as the capabilities and limitations of the concurrent software program became clearer. Based on our increased awareness of the program's capabilities, we decided to also use it in the software start-up process. The program's versatility allowed us to develop a start-up routine that could work on a variety of machines with several configurations while requiring minimal input from the participant.

The time required for programming the concurrent windows and the start-up routine was more than we originally planned. This was because we had very little on which to base our original time estimates, since we had no knowledge of standards for a similar project. We were able, however, to recover this time as we developed the paper-based materials. We selected much of the paper-based materials from the existing course version, and they required only moderate revisions.

Pilot test

Upon completing program development, we conducted a pilot test at the firm's Center for Professional Education. We tested 12 participants—six from U.S. offices and six from offices abroad. All of those from foreign offices spoke English, but it was a second language for four of them. This diversity allowed us to test for both cultural differences and possible language problems. We evaluated the course at two levels—one that measured reactions, and one that measured learning.

The reaction evaluation consisted of a questionnaire the participants completed at the end of each topic. Figure 2 lists the statements participants were asked to respond to; possible answers for all but the last four statements were: strongly agree, agree, uncertain, disagree, and strongly disagree. The questionnaire measured the participants' reactions to each topic on a scale of 1 to 5, 5 being high. Questions addressed the instructional integrity of the topic, the pacing and timing of the topic, and the applicability of the information and skills taught in the topic to

---

Figure 2 - Reaction evaluation

The objectives made it clear what I was to learn.

The content in this topic supported the stated objectives.

At the end of the topic, I felt as though I could achieve the stated objectives.

The sequence in which the information was presented was appropriate.

The level of detail of the content seemed appropriate at this point in my career.

The material in the tutorial was presented in an understandable manner.

The text material was written in an understandable manner.

The examples in this topic were helpful in explaining the concepts presented.

The visuals (charts, diagrams, etc.) helped me understand the concepts presented.

The activities and self-check provided appropriate practice for the concepts in this topic.

The material in the tutorial and the unit guide prepared me for the self-check.

The self-check questions enabled me to determine if I could meet the stated objectives.

The amount of time spent on the activities and self-check in this topic was:
A. About right   B. Too much   C. Too little

The time spent on this topic was:
A. About right   B. Too much   C. Too little

The topic was:
A. Difficult   B. Slightly difficult   C. Neutral
D. Slightly Easy   E. Easy

OVERALL RATING—
Please give this topic an overall rating using the scale below.
A. Excellent   B. Good   C. Fair   D. Marginal   E. Poor

the participants' careers. The questionnaires also gave participants a place to record the amount of time it took them to complete the topic. These times were averaged to give an estimated time per topic, per segment, and for the course.

When evaluating the results of the reaction questionnaire, we cross-tabulated them with a background questionnaire each participant completed before the pilot test. Figure 3 lists the questions and possible answers in the background questionnaire. This helped us review responses based on prior education, prior experience, and English-language proficiency. The overall ratings for the course were very positive; segment scores ranged from 4.08 to 4.22, with 5 being the highest possible.

The learning evaluation consisted of proficiency tests taken by the participants after completing each segment of the course. The average test scores were well above our standard for mastery of self-study material for each of the five segment tests; the scores ranged from 86 percent to 90 percent.

Although both evaluations were positive, we identified a number of necessary revisions. Twenty percent of the resources spent thus far on the project were included in the budget for these postpilot test revisions. This allowed adequate resources to implement the revision points collected and approved during the pilot test.

Success at last

While we are very satisfied with the new fundamentals of accounting course, we are especially pleased to have found an approach to CBT that is cost effective and still allows us to meet all the training goals and objectives of our courses. Based on internal cost standards for the development of CBT, this new approach has allowed us to develop the course for 20 percent of the cost of custom development.

This project has proven that there is viable solution to the dilemma of custom development versus purchasing a less-than-perfect CBT package. Although this approach may not be applicable in all cases, where it does apply it saves money, effort, and time.

Figure 3 — Background

1. Did you take any accounting course(s) in your university studies?

Yes	Yes
A	B

2. How many months of experience do you have with the firm?

0–3	4–6	7–9	More than 9
A	B	C	D

3. How many months of experience have you had with the audit division?

None	1–3	4–9	More than 9
A	B	C	D

4. How many months of experience do you have with the consulting division?

0–3	4–6	7–9	More than 9
A	B	C	D

5. How many months of work experience did you have before joining the firm in areas directly related to consulting division practice?

None	Less than 6 months	6 months to 1 year	1 to 2 years	More than 2 years
A	B	C	D	E

6. How do you evaluate your English-language reading proficiency?

Good	Fair	Poor	English is my native language
A	B	C	D