

Suzuki,K(1987)「A Short-Cycle Approach to CAI Development: Three-Stage Authoring for Practitioners. *Educational Technology*, 27(7), 19-24



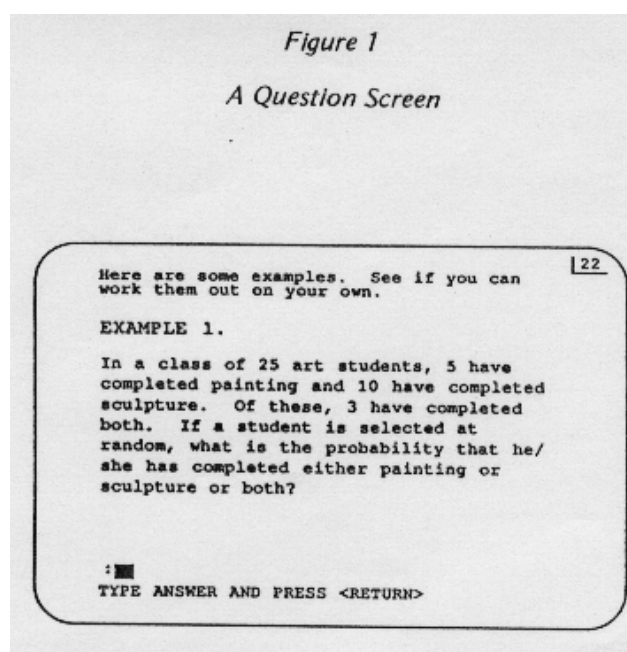
## A Short-Cycle Approach to CAI Development: Three-Stage Authoring for Practitioners

Katsuaki Suzuki

**Katsuaki Suzuki** is a doctoral candidate in Instructional Systems and a Research Assistant at the Center for Educational Technology, Florida State University. The author wishes to thank Dr. Walter Dick for his comments on an earlier version of this article, and Mr. Robert Smith for his editorial assistance.

Although instructional theories and research studies have been demonstrating the effectiveness of CAI, there is less than desired utilization of microcomputers in today's public school education. Given microcomputers as one of the available candidate media for instruction, What can be done to make the best use of this tool? It has long been claimed that microcomputers are capable of being stand-alone instructional media with which individually adapted instruction can be delivered. With the assistance of high quality CAI, classroom teachers are better able to provide effective instruction for each of their pupils, playing the role of instructional managers and other important functions that can best be performed by human beings.

The advancement of authoring programs has made CAI courseware development approachable by classroom teachers, in that a novice author can develop CAI courseware in a relatively short period of time. Typically an authoring program consists of two or more screen display "templates," which automatically provide an author with the basic units of the program. The author needs to do nothing but specify text to show on the screen, questions to ask, feedback messages to give, etc., which are necessary for instructional design for any kind of media. Figure 1 shows a question screen produced using a question-response-feedback template of an authoring program called "Super Softcrates."<sup>1</sup> This particular template allows an author to present a question, to Specify multiple correct answers, to decide the number of tries allowed and feedback message for each wrong response, and to specify a branch to an additional screen after a wrong response.



After individual screens are created as the basic units or components of courseware, the author can then assemble these basic units in an interactive way. The assembly function of the "Super Softcrates" is shown in Figure 2, in which the order of the presentation is specified to form courseware. In this respect, CAI authoring is becoming more and more user-friendly, thus facilitating the creation of CAI courseware by practitioners.

*Figure 2*  
*Assembly Function of an Authoring Program*

Lesson Creator				
LES SON	DI SK	DET AIL	LESSON CONTAINS PAGE DESCRIPTION	34 PAGES
18	51	EX2-5	EDUCATION	
19	60	RULE3	EDUCATION	
20	63	RULE3	- 1	
21	66	RULE3	- 2	
22	69	EX3	- 1 EDUCATION	
23	72	EX3	- 2 EDUCATION	
25	75	EX3	- 3 EDUCATION	
24	78	EX3	- 4 EDUCATION	
26	81	EX3	- 5 EDUCATION	
27	84	RULE4	EDUCATION	
28	87	RULE4	- 1 EDUCATION	
29	90	EX5	- 1 EDUCATION	
31	93	EX5	- 2 EDUCATION	
30	96	EX5	- 3 EDUCATION	
32	99	EX5	- 4	
33	57	EX5	-5 EDUCATION	
34	54	CLOSURE	PAGE	

(A) D (I) NS (D) EL (L) VL (S) EE (B) R (E) X

However, what has been simplified is only the material development, or programming, phase of the CAI authoring process. According to the systematic models of instructional development, considerable amount of time and effort is still required for the front-end analysis and design of courseware prior to programming of the first screen display (e.g., Gagne, Wager, and Rojas, 1981). Further, subsequent formative evaluation and revisions are yet needed to examine and improve the effectiveness of the courseware being developed (Dick and Carey, 1985). If courseware were developed without going through the design and formative evaluation processes, it would be likely that the students would have more difficulty in using the courseware. Since a classroom teacher may not be present when the courseware is to be used, it is important to apply a systematic model of instructional design to CAI courseware development to make the end product useable and effective.

Use of a model of systematic design and development is recommended, but it may require extensive training to be able to adequately employ the model. Moreover, it may take too long for classroom teachers to carry out every step in the model as usually specified. In the remainder of this article an alternative approach to creating effective CAI courseware is proposed which may facilitate the on-site development of the CAI by those who are actually delivering instruction.

### **Basis for the Short-Cycle Approach**

Since differences exist between outside instructional developers and classroom teachers, the conventional process of systematic design and development may be less suitable for classroom teachers than for instructional developers. For the outside development agencies, the formative evaluation process is very expensive, and an opportunity to use the target audience is limited. On the other hand, classroom teachers have daily access to the target audience, but they can not afford the time for extensive design and development of instructional materials. Thus, the existing model fits the situation of the instructional development agencies, where everything should be prepared to the maximum level when formative evaluation is conducted. Classroom teachers, however, can make use of their pupils along the path of design and development, but they need a shorter cycle from the design to the utilization of the materials.

One basis for the proposed short-cycle approach for CAI courseware development can be found in the formative evaluation literature that advocates earlier uses of formative evaluation in the developmental process. For example, although the 'formative evaluation box' is placed at the end of the process in their model, Dick and Carey (1985) suggest formative evaluation of test directions and test items soon after

writing a test. Dick (1977) has also proposed that formative evaluation be applied to each component of the systems development model. Dick's premise was to improve the final product by obtaining formative data for earlier stages of instructional development including goal identification, instructional analysis, entry behaviors and characteristics, objectives, criteriareferenced test, and instructional strategies.

Based on Dick's suggestion, Martelli (1979) has developed, implemented, and evaluated a new model of formative evaluation, in which formative data were collected and utilized to revise the earlier outputs of instructional development. Martelli's findings were limited in that the new model did not produce more effective material compared with the use of the original Dick and Carey model. However, Martelli has shown an alternative way to utilize formative evaluation which may fit the situation of classroom teachers who require a shorter cycle of material development.

Another basis for the proposed approach is that there are several types of CAI courseware. Each type of courseware may be developed by adding or rearranging components of a more basic type of courseware. First of all, a microcomputer can be used as a computer-assisted testing (CAT) system. After a period of usual classroom instruction, a teacher can assign students to terminals to take an end-of-lesson quiz. A set of test items can be programmed by using a question-response-feedback template of an authoring program with no feedback given. Furthermore, if instructional feedback is added to a CAT program, then it becomes drill and practice courseware. In order to keep the assessment function, a parallel set of test items can be developed with feedback which is to be used as the drill part. This would be used for providing individual practice oportunkies after an instructor-ted classroom introduction to the learning task.

Finally, the tutorial-type courseware can be developed by simply adding some information presentation frames and interactive examples using a text template and a question-response-feedback template. If the tutorial part is separated from the practice and testing parts through a menu-driven structure, then all of the three functions will be available for a teacher to use selectively depending upon the student's needs. The relationship among these three basic tyees of courseware is shown in Table 1.

*Table 1*

*Three Types of CAI Courseware to Be Cumulatively Developed Using Authoring Program Templates*

Cycle	Type of Product	Design/Development
1	Computer-assisted Testing Program	A set of criterion-referenced test items using a question-response-feedback template with no feedback.
2	Drill and practice Courseware	A parallel set of practice items added to the testing program via a menu structure, using a question-response-feedback template.
3	Tutorial Courseware	Information presentation frames using a text screen template, added to the drill courseware; with interactive examples utilizing a question-response-feedback template.

### The Short Cycle Approach

Based on the characteristics which are bound to the systematic development of CAI courseware, the proposed short-cycle approach to CAI development consists of three cycles of design, development,

evaluation and revision. Each cycle is short in terms of time required to produce a formatively evaluated end product. Each cycle can be seen as a complete development process, yet can be linked to each other by building one cycle on the product of the previous cycle. The three cycles are described below.

### **Cycle 1 : Computer Assisted Testing Program**

The first cycle is the development of a computer-assisted testing (CAT) program. When need for additional instructional material is perceived by a classroom teacher, the first thing to do is to define what is to be learned. Assuming unfamiliarity of the classroom teacher with systematic design procedures, developing test items, not a learning objective, seems to be the best place to start. Later on, the objective can be derived *from* the test items, which would clarify the process of designing instructional strategies. Using a practice-response-feedback template of an authoring program, developing a CAT program is no more than typing in the questions and specifying the correct answers. Converting an existing test on paper into a microcomputer-based testing program may make the scoring and data keeping processes easier. A text screen template can be used to add a title page to the program.

Soon after the test items are typed in, the program is ready for formative evaluation. A couple of pupils who have already learned the target task can be asked to participate in order to start the clinical, one-to-one formative evaluation process. Troublesome items, typos, etc., will be identified and revisions should take place accordingly. Thus, the use of CAI courseware seems to have become feasible starting from the beginning of formative evaluation, although some (e.g., Golas, 1983) have recommended the use of handwritten frames for the sake of efficiency. The advancement of authoring programs has made courseware revision very easy, thus the use of a paper version may no longer be necessary.

After the revisions have been made based on the one-to-one evaluation, a class of pupils that has just learned the task in teacher-led instruction uses the program as the posttest. This will indicate if the testing program reflects what is taught in the classroom. Common errors that the pupils make should be recorded to be used for the instructional design revision effort in the subsequent cycles. It may also be used as a pretest by a class of pupils that has not yet learned the task. If the results of pretesting reveal that the pupils can do well without any instruction, then the test itself may contain cues, or the learning task may be too easy for the target audience. In-depth discussion of the conduct of formative evaluation can be found in Chapter 10 of Dick and Carey (1985).

### **Cycle 2: Drill and Practice Courseware**

The second cycle of the proposed approach is to develop drill and practice courseware. Since the end product of the first cycle represents the front-end analysis of the instructional program, the design process for this cycle can be simplified to the design of parallel forms of the test items and instructional feedback messages. Congruency among the assessment and instruction is maintained by utilizing the test items as the basis for creating practice items. The errors made by the participants of the first formative evaluation should be used to prepare the distinctive instructional feedback for the wrong responses. It is wise to restrict the anticipated wrong answers to the ones that the target audience is likely to make, and not to try to include unique remedial feedback for every possible type of wrong answers. Using the question-response-feedback template, drill and practice courseware can be developed which may employ a menu structure that has practice and testing as options.

The development of the drill and practice courseware is followed by formative evaluation of the courseware. For this courseware, the possible target audience is two-fold: those who have been exposed to a teacher-led introduction to the target task, and those who have not. The former group can be used to determine if the courseware successfully provides effective practicing events, whereas the latter group will help determine the extent of further development that is necessary. If the pupils can learn the target task without initial instruction to the task, then the drill and practice courseware by itself may be adequate for that particular task, and shifting to develop courseware concerning other tasks may be more beneficial. Since an extensive needs assessment has not been conducted at the outset of the first cycle, data from the target audience can be used to monitor the need for further courseware development.

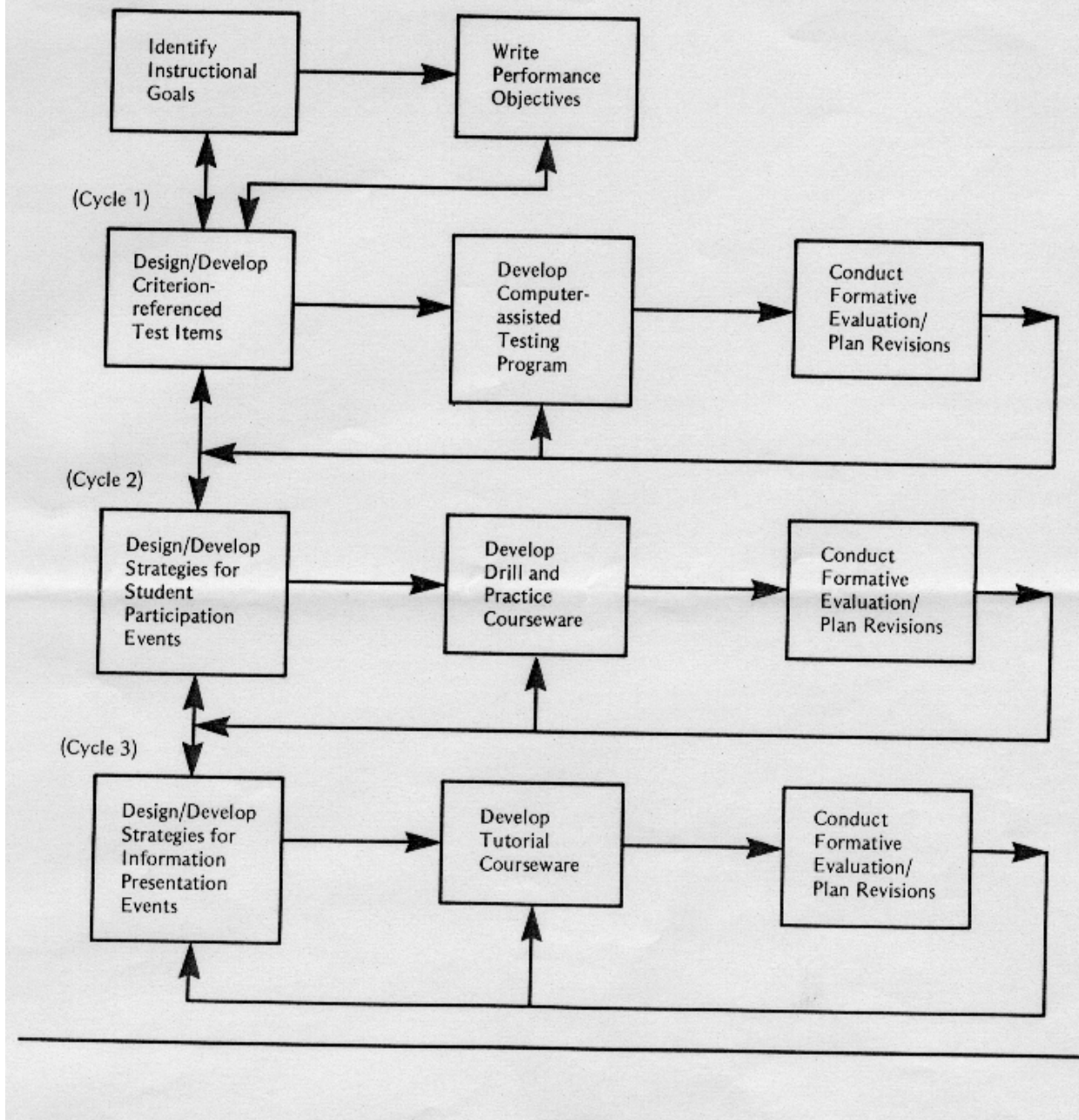
### **Cycle 3: Tutorial Courseware**

The third and final cycle of the development is tutorial courseware. Based on the need for further material development for the target learning task, or to eliminate the classroom introduction to the target task, a set of information presentation screens and interactive examples will be added to the drill and practice courseware (the end product from the second cycle of the development). At this point, a closer look at instructional theories as applied to CAI design (e.g., Gagne et al., 1981) may be helpful to derive effective instructional strategies in the information presentation section.

Although the term 'tutorial' is used to describe this cycle, the courseware to be developed in this cycle may take diverse forms, depending on the results of the formative evaluation in the second cycle. For example, it may be just creating optional HELP sequences attached to the drill and practice courseware, it may be just adding a summary page about the learning task before the drill, or it may be developing another section from the menu page devoted to the introduction of the task with a number of shaping exercises including some sub-ordinate prerequisite skills. A text screen template can be used to present information, and various question-response-feedback templates can be utilized for interactive examples and basic exercises.

The final formative evaluation on the tutorial courseware will then be conducted in a conventional way to determine which parts of the tutorial courseware to revise. The revision of the courseware, however, may be different from when a conventional model is followed because a part of the courseware has already been revised based on the data from the target audience in the earlier cycles. Thus, if performance data show a need for revision, it is most likely that the problem area is not in practice nor testing, but in the tutorial. More explanation of the task may be necessary, or more guidance that makes the learning task meaningful may need to be added in the tutorial part of the courseware. The formative evaluation of the tutorial courseware and subsequent revisions complete the development of CAI courseware using the short-cycle approach. The overall process of the approach is summarized in Figure 3.

Figure 3

*Proposed Short-cycle Approach to  
CAI Courseware Development***Advantages of the Short-Cycle Approach**

The final product of this approach is stand-alone tutorial CAI courseware, which has been formatively evaluated by the target audience. Also developed are a computer-assisted testing program and drill and practice courseware, both of which can be used as parts of the tutorial program. Thus, a classroom teacher who has been the developer of the courseware can assign the courseware to students and have them work through the courseware independently to learn new skills or knowledge, to practice the previously learned skills, and/or to test the skills. Although the final product may be the same as what existing models of instructional design and development produce, the proposed short-cycle approach can be adopted with some advantages particularly when used by instructional practitioners such as classroom teachers.

One of its advantages lies in its flexibility regarding time schedules. Following an existing systems

model, the entire course of design, development, and evaluation needs to be completed before the final product will be available for use. If the process is terminated at any point before the final product, then no product which is ready to be used would be available. The proposed approach, however, consists of three stages of development, each of which represents a short cycle of design, development, and evaluation. Thus, the process may be ended at any stage of development depending on available resources and the need for the product, still resulting in an effective and usable intermediate product.

This flexibility may also be seen as allowing for changing requirements for the material being developed. For example, after trying out the drill and practice type courseware, there may not be a continuing need for extensive tutorial courseware if everybody in the class can learn the task by being exposed to a relatively short instructor-led introduction followed by the drill and practice. What is needed may not be a tutorial, but only the drill and practice courseware. When drill and practice courseware needs to be developed, results from the formative evaluation of the testing program can be used to anticipate plausible types of wrong responses. For example, the types of wrong responses can be identified for which corrective instructional feedback may best be utilized. Thus, the principle of 'lean programming' will be applied both to the type of the courseware and to the instructional strategies used in the courseware.

If the short-cycle approach is followed to develop tutorial courseware, the congruency among the test, practice, and tutorial is likely to be high due to the built-in evaluation mechanism. Use of the parallel forms of the test items in drill and practice type courseware will ensure that the test and practice items match each other. If the tutorial is designed around the practice items, the information presentation may well become more direct and straightforward. As a consequence, not only the effectiveness of the courseware but also the efficiency may well be improved as the result of the short-cycle approach.

Concise and effective courseware will also help making the courseware appealing to the students, motivating them to use the CAI courseware without the instructor's frequent assistance. The suggested approach will facilitate employing effective motivational strategies (Keller and Suzuki, *in press*) such as a menu-driven structure, instructional feedback, and the congruency among the components of the courseware.

A final advantage can be found in the direct connection between formative data and subsequent revisions. In the first stage of developing computer assisted testing, major deficiencies will be taken out from the courseware regarding the test items. In the next stage of drill and practice development, problems would show up in practice items or feedback messages, not in the test items. By the time the tutorial courseware is developed the information presentation part would be more likely to contain some problems. Since each of the cycles represent fewer interventions than in the conventional models, interpretation of the formative data will become easier, which may facilitate more 'appropriate' revisions.

## Conclusion

In sum, this article describes an alternative approach to the systematic design and development of CAI courseware, especially for those who are closely working with the potential audience of the courseware. The short-cycle approach consists of three main stages of courseware development, each of which produces a useable output (computer-assisted testing program, drill and practice courseware, and tutorial courseware, respectively). It is the author's hope that this alternative approach can be utilized by instructional practitioners to make the best use of the microcomputer as an instructional medium, as well as an efficient method of applying the principles of instructional theory to practice.

## Note

1. "Super Softcrates" is a product of Simpac Educational Systems, Suite 11-C, 1105 North Main Street, Gainesville Florida 32601.

---

## References

Dick,W. *Application of Formative Evaluation to the Insutructional Design Process*. Paper presented at the annual meeting of the American Educational Research Association, New York, April 1977.

Dick,W., and Carey, L. *The Systematic Design of Instruction* (2nd Ed.). Glenview, IL: Scott, Foresman, and Company, 1985.

Gagne,R.M., Wager, W., and Rojas, A. Planning and Authoring Computer-Assisted Instruction Lessons. *Educational Technology*, September 1981,21(9), 17-26.

Golas,K.C. Formative Evaluation of Computer-Assisted Instruction. *Edecational Techonology*, January 1983, 23(1), 26-28.

Keller,J.M., and Suzuki,K. Use of the ARCS Motivational Model in Courseware Design. In D.H.

Jonassen(Ed.), *Instructional Designs for Micorocomputer Courseware*. Hillsdale, Nj: Lawrence Erlbaum,in press.

Martelli,M. *A Study of a Theory-Based Model of Formative Evaluation*. Unpublished dectoral dissertation, Florida State University, 1979.

