DOCUMENT RESUME

ED 289 501 IR 012 968

AUTHOR	Ellington, Henry
TITLE	How To Produce Computer-Based Learning Materials. Teaching and Learning in Higher Education, 18.
INSTITUTION	Scottish Central Institutions Committee for Educational Development.
SPONS AGENCY	Robert Gordon's Inst. of Technology, Aberdeen (Scotland).
PUB DATE	87
NOTE	21p.; For related guides, see IR 012 959-960.
PUB TYPE	Guides - Non-Classroom Use (055) -- Information Analyses (070)
EDRS PRICE	MF01/PC01 Plus Postage.
DESCRIPTORS	Authoring Aids (Programing); *Computer Assisted Instruction; Computer Managed Instruction; Computer Simulation; *Courseware; Foreign Countries; Higher Education; Instructional Design; *Interactive Video; *Material Development; *Models; *Production Techniques; Teacher Developed Materials; Teaching Methods

ABSTRACT

A sequel to the booklet "How Computers Can Be Used in Education," this booklet begins by looking at ways in which computer-based learning (CBL) materials can be used in different types of instructional situations, i.e., mass instruction, individualized learning, and group learning. The design and production of four types of conventional CBL instructional materials are then examined: (1) number crunching (calculation) and data processing packages; (2) substitute tutor packages; (3) substitute laboratory packages; and (4) computer-managed learning packages. Guidelines for the production of interactive video materials conclude the booklet. Sample materials provided include an extract from a typical tutor-mode CBL dialog; models of drill and practice, tutorial, and simulation lessons; and a diagram of the features of an interactive video system. An annotated list of six items recommended for further reading is included. (MES)

This booklet was first published internally in Robert Gordon's Institute of Technology, Aberdeen as part of the Institute's staff development programme.

The booklet was written by Dr Henry Ellington of RGIT's Educational Technology Unit, who adapted it from material originally published in "Producing Teaching Materials", by Henry Ellington (Kogan Page, London; 1985).

CICED gratefully acknowledges the co-operation of RGIT, Kogan Page and the author in the publication of the present edition of the booklet

How to Produce Computer-based Learning Materials

Introduction

This booklet is a sequel to booklet number 9 in the suite – "How computers can be used in education". In this other booklet (which it is recommended should be read *before* the present booklet), we examined the various roles that computers are capable of playing in education, and discussed some of the main factors that are likely to affect their use. In the present booklet, we will turn our attention to the production of computer-based learning (CBL) materials, showing how materials of the various types identified in the earlier booklet can be developed by practising teachers and lecturers.

As in the other booklets in this series that deal with the production of instructional materials, we will begin by taking a general look at how computer-based learning materials can be used in the three different types of instructional situations that were identified in "A guide to the selection of instructional methods" – mass instruction, individualised learning and group learning. We will then turn our attention to the design and production of CBL materials, looking first at the development of 'conventional' CBL materials and then at the development of interactive video materials.

How computer-mediated materials can be used in different teaching/learning situations

In "A guide to the selection of instructional methods", it is argued that teaching/learning methods can be divided into three broad classes – *mass instruction techniques, individualised learning techniques and group learning techniques*. Let us now take a general look at how computer-mediated materials can be used in each of these different types of instructional situation.

Mass instruction

To date, the main use of computers in mass instruction has been in a supportive role during conventional expository lessons rather than as a vehicle for mediated exposition, although current developments in the field of interactive video may well cause this situation to change. In such a supportive role, computers can be used in a wide variety of ways, e.g.:

- As vehicles for teaching *about* computers and teaching computer programming;
- As a means of carrying out complicated calculations or data processing activities in the course of a lecture, lesson or training session, and of demonstrating these to the class;
- As a vehicle for demonstrating simulations of all sorts to a class;
- As a means of generating graphical materials and demonstrating these to a class;
- As a mode of entry to data bases of all types.

Individualised learning

Although computers are capable of playing an extremely useful role in mass instruction, their most important potential role in education and training is almost certainly in the area of individualised learning. Here, 'conventional' computer-based learning and interactive video systems probably constitute the most powerful delivery systems yet developed, and, in the opinion of many commentators, seem likely to bring about changes in education and training that can only be described as revolutionary (see booklet on "How computers can be used in education"). Some of the main ways in which computer-mediated materials can be used in individualised instruction are outlined below:

- *Use as a 'substitute tutor'*. Here, the learner is guided through an instructional sequence of some sort by carrying out an on-line dialogue with a computer via an interactive terminal (usually a keyboard/video display unit or interactive video work station). The computer is programmed to present the learner with information, ask questions, and react to the responses by presenting further information or questions. This adaptive style of learning is directly descended from the *programmed learning* movement of the 1950's and 60's. It is essentially similar to 'branching' programmed learning, but is capable of being much more sophisticated that the latter because of the greater flexibility and data-handling capacity of computer-based learning and interactive video delivery systems compared with conventional teaching machines and programmed texts. Tutor-mode computer-assisted learning (CAL) of this type seems certain to play an increasingly important role in education and training during the remainder of this century, and may well bring about the massive swing from conventional expository teaching to mediated individualised learning that some commentators are currently predicting (see "How computers can be used in education").

- *Use as a 'substitute laboratory'*. Here, the computer or interactive video system involved is more of a learning resource than a direct instructional device. In this mode, real-life or hypothetical situations of all types can be modelled on the computer or computer-video system, thus allowing the effect on the situation of changing key variables or parameters to be studied by the learner. In this way, learners can be given experience of a far wider range of situations than would ever be possible by conventional means, and can, furthermore, do so in an enactive rather than a passive learning situation – something that always greatly increases the effectiveness of an instructional process. Computer simulations of this type also seem certain to play an increasingly important role in future education and training.
- *Use in a managerial or supervisory role*. Here, the computer acts as a manager or controller of the learning process rather than making a direct contribution to the teaching/learning process, as is the case in computer-assisted learning. In other words, it acts in a supportive and/or supervisory role, relieving the teacher or trainer of some of the more tedious or time-consuming tasks normally associated with individualised learning, thus allowing him to devote more time to meeting the needs of individual learners. Specific ways in which computers can contribute to computer-managed learning (CML) include administration and marking of tests of various types, providing on-going guidance to learners based on their performance and individual needs, and maintaining up-to-date records of the progress both of individual learners and of the student/trainee body as a whole.

Group learning

Computers can also make a significant contribution to a wide range of group learning activities. They can, for example, be used to manage or structure a group learning process, e.g. by guiding the group through a simulation exercise of some sort. They can also provide a vehicle through or with which a group of learners can interact, as well as providing facilities for gaining access to data bases, performing calculations, investigating simulated situations, generating graphics, etc.

How to design and produce 'conventional' CBL materials

Let us now turn our attention to the production of computer-mediated instructional materials, starting by looking at how to design and produce 'conventional' CBL materials – to distinguish them from CBL materials of the interactive video type. We will look in turn at the production of four different types of material – *'number crunching'* *and data processing packages, 'substitute tutor' packages, 'substitute laboratory' packages,* and *computer-managed learning packages.*

How to produce 'number crunching' and data processing packages

The original reason why computers were developed was in order to provide help with complicated calculations and data processing tasks, and this remains one of their most important roles – in all types of instructional situations. Certainly, this is an area where computers can be of considerable assistance to college lecturers – particularly those who work in disciplines (such as science, engineering or economics) that involve a lot of calculation or data handling. Computers can, for example, be used to carry out standard calculations and data processing activities (such as the calculation of the means and standard deviations of sets of raw data and the determination of the slopes of graphs) by using appropriate sotware pachages. They can also be used to process data that is fed directly into them from a piece of equipment or apparatus via a suitable interfacing system – a technique that is finding an increasing number of applications in research and teaching laboratories of all types. For example, a useful tool in physics courses is an interface system and software package that enables a Geiger counter to be connected directly to a microcomputer. This enables students to carry out certain experiments on radioactive decay, absorption of ionising radiation, etc. without having to spend hours tediously accumulating raw data. The computer now does this for them virtually automatically, thus enabling the students to proceed fairly rapidly to the *interpretation* of the data – the part of the work from which they derive most educational value.

If you are interested in developing a 'number crunching' or data processing package of this type , the way in which to set about the task will obviously depend on a number of factors, including:

- the nature of the calculation or data processing activity to be carried out;
- the nature of the computer to be used (mainframe, mini or micro? make? type of peripherals available? etc.);
- the extent of your programming ability and experience.

If the calculation or processing activity is a fairly simple one, and you are confident that you possess the necessary programming skills, the best way to proceed is probably to 'start from scratch', and write the entire package yourself. If the job is a difficult or complicted one, on the other hand, or if your programming skills or experience are limited, it will probably pay you to seek advice and/or help – either from a colleague who has more experience of programming than you or from a professional programmer. Such a person may well be able to direct you to an off-the-shelf software package that will be able to do the job you have in mind, either as it stands or after suitable modification. Failing this, he will almost certainly be able to give you a considerable amount of assistance in the production of the sort of package that you want. The services of professional programmers are now becoming increasingly available to teachers and trainers of all types – either within their own establishments or in outside units (such as Teachers' Centres) to which they have acess – and you should never hesitate to seek their help; it could save you a great deal of time and effort.

When designing a package of the type being considered, it is essential to make it as 'user friendly' as possible – particularly if it is to be used by people who have little knowledge or experience of computers. If at all possible, incorporate any operating instructions in the actual program, so that they are presented automatically when the user switches on the machine and/or calls up the program. In many cases, it is also a good idea to provide the user with a hard copy printout rather than simply a soft copy readout on a VDU screen, but this will obviously depend on the nature of the hardware that is at your disposal.

Once you think that you have got the program working properly, and feel that you have trapped all conceivable errors, it is strongly advisable to ask a colleague to give it a thorough work-out *before* proceeding to the field testing stage. Even experienced programmers sometimes fail to spot errors in their own programs, and are often so close to their work that they are not capable of spotting flaws that only become apparent when someone who is not familiar with the system does something silly. Asking a colleague (or a member of your family) to see if they can find a way to crash your program is often a very effective way of bringing such flaws to light!

How to produce 'substitute tutor' packages

In the type of packages described in the last section, the computer is simply used as a tool to help carry out a specific task – performance of calculations or processing of data. In 'substitute tutor' – mode CBL, on the other hand, the computer takes a much more active role, effectively controlling the process by which the learner is given personal tuition on a particular topic or guided through an instructional or decision-making sequence of some sort. A typical example of an interactive dialogue of this type is given in figure 1.

Here, the computer is helping a chemistry undergraduate to plan an experiment in physical chemistry. In the dialogue, the student responses are italicised.

You are required to investigate experimentally the chemical equilibrium of the system: $Fe^{3} + SCN \rightleftharpoons FeNCS^{2}$ in order to discover how the equilibrium constant varies as a function of the ionic strength. You need a method which is both sensitive and discriminating. Which method(s) do you suggest?

I'm sorry, but I haven't a clue

You first need to consider the physical properties of the individual species, or of the system as a whole, which you might use.
What feature do all the species involved have in common?

They are all ionic

Good!
So what method or methods might you choose which makes use of this property?

Conductivity methods

Yes - a conductance method.
Is a conductance method sensitive? (Very, Fairly, Not Very, Not at All)

Very

No. A conductance method is only fairly sensitive.
Is it a discriminating method?

Yes

Conductance methods are only really discriminating when applied to H+ ions: for all other ions, the readings are rather similar.
Do you want to consider this method further?

No

What other method(s) might be more discriminating and sensitive?

Possibly a colourimetric method

Good. Are colourimeric methods sensitive?

and so on.

Figure 1: extract from a typical tutor-mode CBL dialogue

The design of tutor-mode CBL materials - use of authoring systems

There are two basic ways in which it is possible to approach the task of designing a tutor-mode CBL package. The first - and most obvious-method is to design the instructional process first and then produce a computer program that can be used to administer this instructional process. If you possess the necessary programming skills (or have access to the services of someone who does) then there is no reason why you should not be able to produce perfectly acceptable CBL materials in this way, although it will probably take you a relatively long time to do so.

The second method is to make use of a suitable *authoring language or authoring system* to help you produce the material. These are specially designed so as to enable people *with little knowledge of computer programming* to produce CBL materials by providing a variety of standard structures within which the materials can be developed, and undoubtedly constitute the easiest and most cost-effective method of developing tutor-mode CBL materials. A wide range of such authoring languages and systems are now available, for use on virtually all types of computer. Detailed information of those that were available in 1983 are given in Dean and Whitlock's "Handbook of Computer-Based Training" (see 'Further Reading' section), but this is almost certainly hopelessly out of date by now, since new systems are being developed all the time. Thus, any reader who is interested in using an authoring language or authoring system to develop CBL materials should seek the advice of a professional programmer or computer consultant; such a person will be able to direct you to the system that is best suited for your particular needs.

Examples of the sort of facilities that authoring systems provide

In order to show readers the sort of materials that can be produced using a suitable authoring system, details of two of the standard models that are available in one such system - the 'PLATO Stand-Alone Author and Delivery System' developed by Control Data - are given below.

The first, the 'Drill and Practice Model', is shown in figure 2. This has the following three stages.

(i) The learner is shown a title page and introduction (created by the writer of the lesson) and is then given a standard set of instructions on how to use the lesson.

(ii) The learner is set a series of questions created by the writer; these can either be presented in a predetermined sequence or culled at random from a bank of questions.

(iii) The computer determines the percentage of correct answers, and, using criteria set by the writer of the package, directs the learner either to move on to the next stage of the instructional process or to repeat the lesson.

Drill and Practice Model

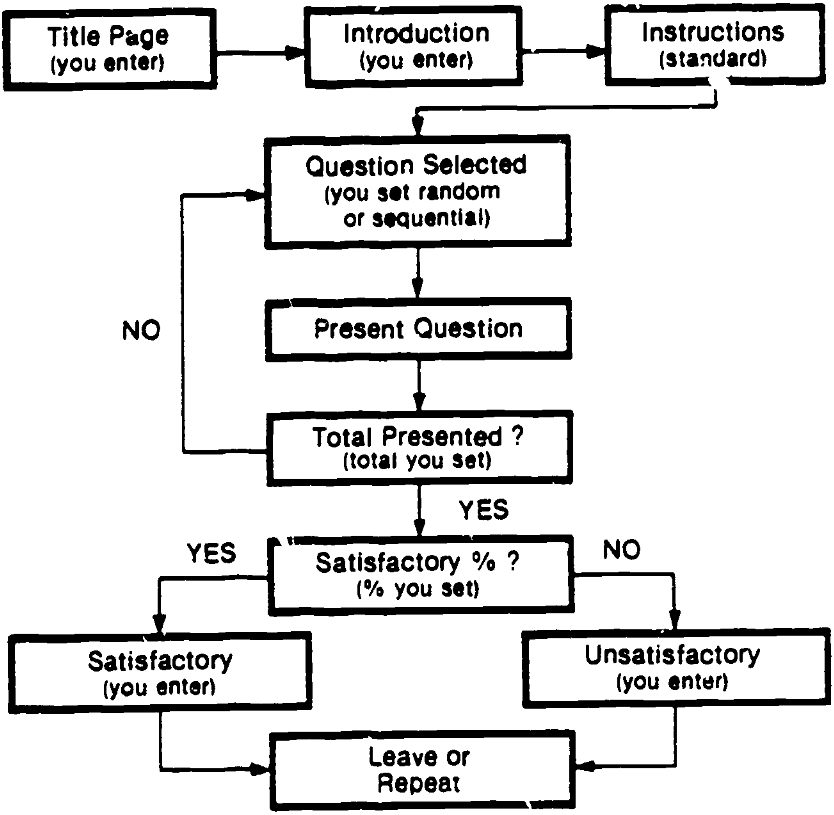


Figure 2: the 'Drill and Practice Model' available in the PLATO authoring system.

A second PLATO Model, the 'Tutorial Lesson Model', is shown in figure 3.

This has the following five stages.

(i) The learner is presented with a brief introduction to the area to be studied, written by the author of the package.

(ii) A menu is displayed, allowing the learner to choose from a number of specific topics in the area in question (again selected by the author).

(iii) Having chosen a topic, the learner is presented with instructional information (written by the author) relevant to that topic.

(iv) The learner is asked a diagnostic question (written by the author) designed to determine whether or not the instructional information has been mastered.

(v) If the answer is satisfactory, the learner is told to proceed to the next topic that he or she wants to study (again chosen from the menu displayed in Stage (ii); if not, the learner is advised to review the lesson material before trying again.

Tutorial Lesson Model

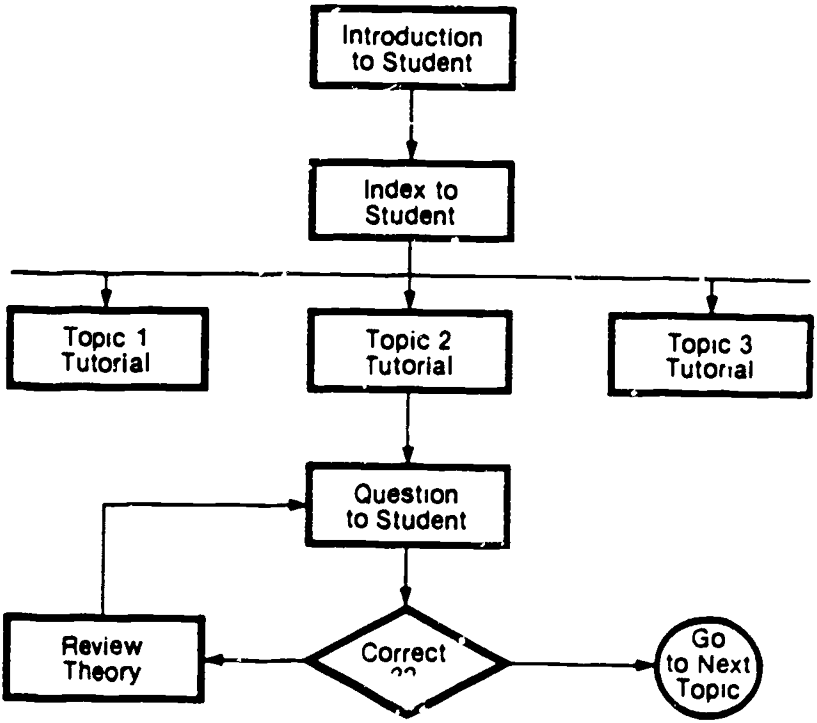


Figure 3: the 'Tutorial Lesson Model' available in the PLATO authoring system

Note that most authoring systems provide facilities for sequences written in conventional programming languages such as BASIC and FORTRAN to be included in the instructional sequence, and that many also allow graphic materials to be incorporated. The exact nature of the facilities available vary considerably from system to system, with the sophistication of the facilities tending to increase with that of the hardware with which it is designed to be used. Thus, systems designed for use with inexpensive general-purpose microcomputers tend to be far less sophisticated than those designed for use on mainframe machines or machines that have been specially developed for CBL work (e.g. the specialised microcomputers used in the PLATO system). Nevertheless, such simple systems are often all that is required to produce perfectly adequate CBL materials. Owners of BBC microcomputers, for example, will probably find that the MICROTEXT authoring system that is available from the National Physical Laboratory is capable of meeting most of their needs.

Readers who are interested in producing tutor-mode CBL materials will find detailed guidance on how to design the materials in the books by Dean and Whitlock and by Godfrey and Sterling that are listed in the 'Further Reading' section.

How to produce 'substitute laboratory' packages

As we have seen, 'substitute laboratory' CAL packages, in which learners are given experience of situations or enabled to investigate systems via the medium of computer simulations, seem likely to play an increasingly important role in education and training. In the various branches of science and engineering, for example, they can be used to provide instructional and training experiences that would simply not be practicable using conventional methods on grounds of cost, time, safety, etc. (e.g. experiments in genetics or work with systems such as nuclear reactors). Computer simulations can also prove extremely useful in the various social sciences, as well as in fields such as business management and commercial, industrial and military training.

The design of 'substitute laboratory' packages

As in the case of 'substitute tutor' CBL materials, there are two basic approaches that can be adopted to the design of a 'substitute laboratory' package. The first is to begin by deciding on the main features of the situation or system of which you want the learners to be given experience, and then to develop a computer model that incorporates these various features using a conventional high-level programming language such as BASIC or FORTRAN. The second is to make use of a suitable authoring system or special simulation language that incorporates a standard framework in which a model of the type you want can be developed, and to work within the context of this.

If you possess the necessary programming skills, and the system to be simulated is a relatively simple one, the first approach may well be the best way to proceed. With more complicated systems, on the other hand, use of a specialised simulation language or authoring system could offer distinct advantages. Again, readers who are interested in making use of facilities of this type are advised to seek the help of a professional programmer or computer consultant.

An example of the sort of facilities that simulation authoring systems provide

In order to illustrate the sort of facilities that are provided via CBL authoring systems, let us again look at one of the standard models that are available in one such system – the 'Situation Simulation Model' from the PLATO system. This model, which is shown in figure 4, enables highly complicated branching decision-making simulations to be developed, and again requires absolutely no programming experience on the part of the writer. As can be seen from the figure, the learner is first confronted with a situation of the type that he might encounter in real life (devised by the writer of the situation) and asked to state what action he would take (from a number of alternatives chosen by the writer). The learner is then confronted with the new situation that would arise as a result of this action, and again asked to state what he or she would do. The process is repeated as often as necessary. Other authoring systems such as MICROTEXT offer similar facilities for the modelling of situation simulations of this type.

Producing supportive courseware for computer simulations

One important difference between tutor-mode and laboratory-mode CBL packages is that the former are generally completely free-standing, with the computer presenting all the relevant information to the learner, whereas the latter usually include supportive courseware of some sort – often in the form of a 'user's manual'. The need for such courseware stems from the different role that the computer plays in laboratory-mode CBL, where (as we have seen) it is more of a learning resource than a vehicle for mediated instruction. Thus, there is often a need to give the user instructions or guidance on how to use the resource, e.g. by suggesting possible 'experiments' that might be carried out using the model round which the package is built. In some cases (e.g. situation simulations of the type illustrated in figure 4) no such supportive courseware may be necessary, but in others (e.g. simulations of physical, biological, social or economic systems with several independent variables) the user's guide may be just as important a component of the package as the computer itself. Thus, great care should always be taken in the planning and design of such a document, since the ultimate success of the package as a teaching tool may well depend on it.

Situation Simulation Model

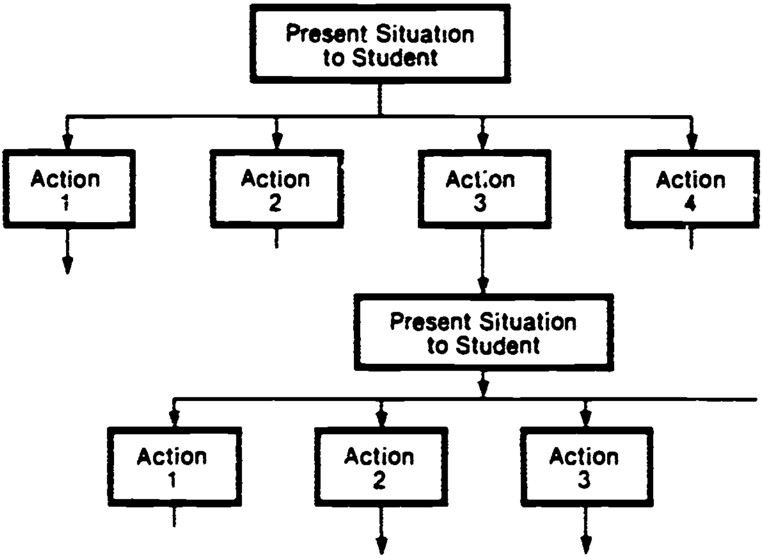


Figure 4: the 'Situation Simulation Model' available in the PLATO authoring system

How to produce computer–managed learning packages

In *computer–managed learning* (CML), we have seen that the computer is used in a managerial or clerical role rather than taking a direct part in the actual process of instruction. Specifically, it can be used:

- to generate, adminster, mark and analyse tests for diagnostic and/or assessment purposes;
- to provide individual guidance to learners about such things as choice of modules, choice of route through a course, and whether or not remedial work is required;

- to maintain an up-to-date record of the progress and performance of all the individual students or trainees on a course, so that a tutor or instructor can see how any particular learner is doing at any time.
- to provide an on-going overview of the progress of the student body as a whole, or on the operation of the course in general, to the people responsible for running and planning a course.

Because of the complexity of even a modest CML system, anyone contemplating making use of such a system would be well advised to try to find an existing software package that is capable of doing the job rather than trying to develop his or her own package 'from scratch'. Needless to say, the advice of a professional programmer or computer consultant should again be sought, since such a person may well be able to recommend a suitable package – or a package that can be adapted to do the job required.

Readers who want to find out more about computer-managed learning are referred to the book by Rushby listed in the 'Further Reading' section; Chaper 4 of this deals with the subject in some detail.

How to produce interactive video materials

It is now generally accepted that interctive video contitutes potentially the most powerful medium for mediated instruction yet developed, combining, as it does, the facilities of computer-based learning with those of television. Using such a system, which interfaces an interactive computer terminal with a random-access videotape or videodisc recorder and a television monitor, the author of a CBL programme is no longer restricted to the use of conventional computer graphics, which, for all the advances made during recent years, are still severely limited in the extent to which they can represent real-life situations satisfactorily. Rather, he or she can build high-quality television pictures into the programme (complete with sound track, if required) so that an appropriate sequence can be called up from the videorecorder by the computer whenever it is needed. Needless to say, such a facility adds a completely new dimension to computer-based learning.

The components of an interactive video system

The essential features of an interactive video system are shown schematically in figure 5.

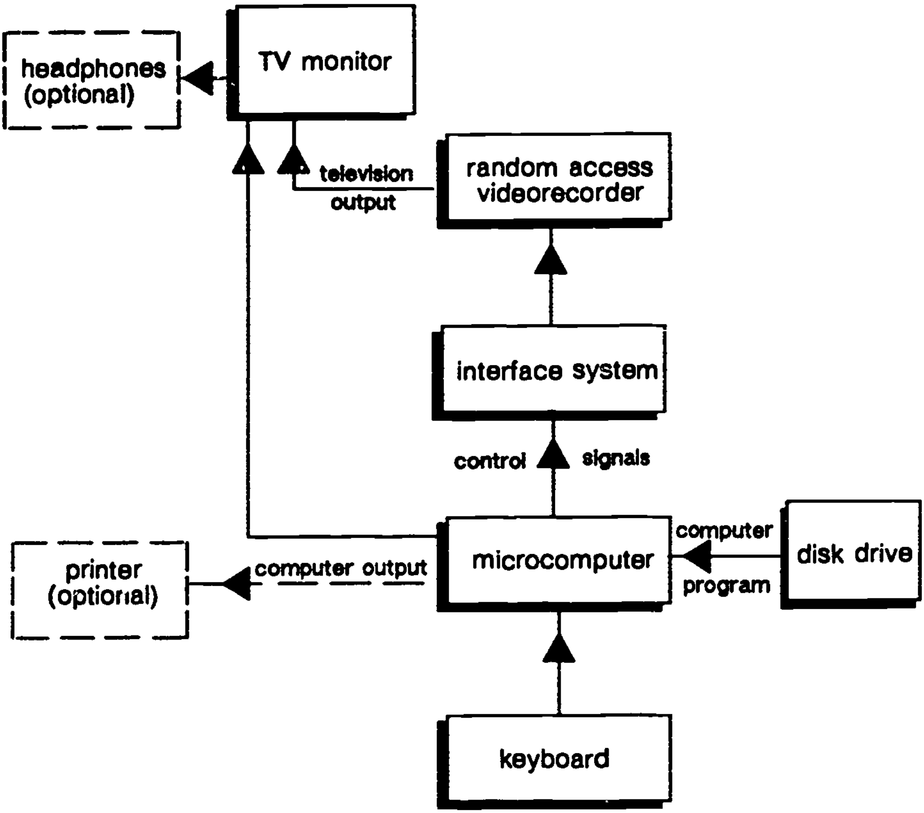
As can be seen, the heart of the system is a suitable microcomputer that is connected to a random-access videorecorder by special interfacing equipment. The videorecorder can be either a videotape machine (usually a special videocassette recorder with twin audio tracks) or a videodisc machine (usually an optical videodisc player). Most early interactive video work was done using videotape, but videodisc-based systems (which are capable of handling much more sophisticated programmes) are now becoming increasingly widely used.

In both types of system, the content of the programme (the *courseware*) is stored on two seperate media – the video component being stored on a videocassette or videodisc and the computer component (i.e. the frames of the CBL sequence into which the video material is built) being held on a floppy disk. The latter also carries the 'system software' (the computer programs that control the presentation of the two types of material to the user). The outputs of both the computer and the videorecorder are fed into a television monitor, usually fitted with headphones if the system is being used for individual study. Some systems also include a printer, which can be used to provide the user with with a hard-copy printout of any of the computer-generated material of which a permanent record is required. In all interactive video systems, the user communicates with the computer via a keyboard or keypad of some sort or via a touch-sensitive monitor screen.

The different methods of producing interactive video materials

At the time of writing, most educators and trainers who want to develop their own interactive video materials 'in house' are more-or-less restricted to the use of a videotape-based system of some sort. Only those who work in a very large organisation or have access to massive external funding can hope to produce videodisc-based materials, since the equipment required is not only extremely expensive but also requires highly specialised technical support for its proper use. Detailed information about what the development of such materials involves can be found in the CET Working Paper (by Duke) that is listed in the 'Further Reading' section, and interested readers are referred to this.

As in the case of conventional CBL materials, the development of videotape-based interactive video materials can be tackled in two ways. If you possess the necessary electronic and programming skills, there is nothing to stop you from connecting a standard microcomputer (such as an Apple II) to a suitable videocassette recorder (e.g. a U-matic or industrial VHS machine) via special interfacing circuits, and converting conventional video sequences



into interactive video programmes using the resulting system. This is probably the cheapest way of producing interactive video materials, but is certainly not the easiest, since fairly intimate knowledge of the system and advanced programming skills are required. Thus, anyone who does not possess such knowledge and skills would be well advised to make use of a custombuilt interactive video authoring facility. A number of systems of this type, which enable people with no knowledge of electronics or programming to develop highly-sophisticated interactive video materials, are now available commercially. One such system – the Computer Audio Video Instruction System (CAVIS) developed by CAVIS-Scicon, is shown in figure 6. This is a dual-purpose facility that can serve both as an authoring system for the creation of interactive video materials and as a work station for the study of such materials by individual students or trainees. The photograph shows the system being used in the former mode; when used in the study mode, the large keyboard shown in use in the photograph is replaced by a simple keypad (seen at the right hand side of the photograph, on top of the control cabinet that contains the videocassette recorder, computer, and interfacing equipment). At the time of writing, a complete CAVIS system cost roughly £14,000.

Figure 5: the different components of an interactive video system

How to produce programmes

Let us now conclude this discussion of interactive video by taking a broad look at what the design and production of an interactive video programme involves.

(i) *The design phase*

This is similar in many ways to the design of a conventional CBL programme, involving the following stages:

- Establishment of the exact instructional role that the programme is to fulfil, including detailed formulation of its design objectives,

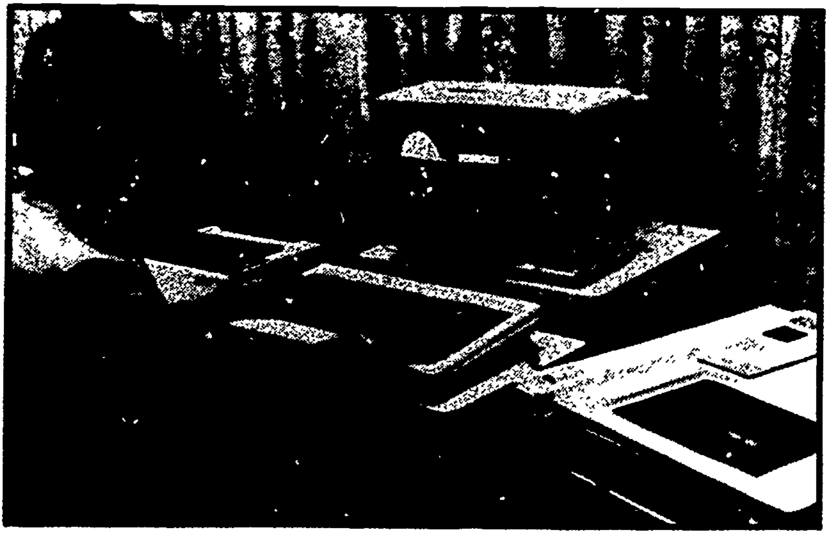


Figure 6: a typical interactive video authoring facility – the CAVIS system

- Deciding on the overall structure and content of the programme and establishment of an outline plan for same.

- Development of the detailed structure of the programme, showing how the CBL and video elements will inter-relate; detailed design of the CBL and video materials.

(ii) *The production phase*

This also involves three stages, namely:

- The production of the video components of the programme and editing of same into the final continuity required; this should be done by the methods described in booklet number 17 in this series – "How to produce video materials".
- The creation of the various frames of the CBL sequence into which the video materials are to be integrated using suitable computer equipment (e.g. a custom-designed interactive video authoring facility such as the CAVIS system).
- The integration of the video and CBL components to form the final programme. As explained above, by far the easiest way to do this is again to use a specialised authoring facility such as CAVIS, which enables video materials to be converted into interactive form by people with no programming skills. Indeed, use of such a system makes the production of the final programme one of the easiest parts of the entire development process!

Readers requiring more detailed guidance on how to develop interactive video materials are referred to the books on interactive video by Duke and by Parslow that are listed in the 'Further Reading' section, and also to the research report by Bryce.

Further Reading

1. *An Introduction to Educational Computing*, by N.J. Rushby; Croom Helm, London; 1979. (An excellent introduction to the field, written for the non-specialist.)
2. *The Elements of CAL*, by D. Godfrey and S. Sterling; Reston Publishing Co., Virginia; 1982. (An extremely useful handbook for anyone interested in producing CBL materials.)
3. *A Handbook of Computer-Based Training*, by C. Dean and Q. Whitlock; Kogan Page, London; 1983. (Another extremely useful basic text on CBL – and one that should prove just as useful to people working in education as to the trainers from whom it is primarily written.)

4. *Interactive Video*, by E. Parslow; John Wiley, Chichester 1984. (One of the first basic texts to appear on the subject – extremely useful to anyone interested in working with the medium.)

5. *Interactive Video – Implications for Education and Training*, by J. Duke; Council for Educational Technology, London; 1983. (An extremely interesting CET Working Paper that explains exactly what is involved in producing high-quality interactive video materials.)

6. *Improved CAI by the use of interfaced random-access audio-visual equipment*, by C. Bryce; Dundee College of Technology Research Report P/24/1; 1982. (An extremely helpful report for anyone interested in carrying out interctive video work by adapting standard video tape and microcomputer equipment.)