

Software for the Leicester Interactive Videodisc project

Clive Ruggles*

37.1 Introduction

The Leicester Interactive Videodisc project, funded by the University Grants Committee under their Computers in Teaching Initiative, aims to investigate the application of interactive videodiscs in the teaching of archaeology to undergraduates. This paper describes progress on the software development for the project.

37.2 Interactive videodisc technology

Interactive videodisc technology is ideal for developing the skills of interpreting visual material which are so fundamentally important in archaeology but which all too frequently can not feasibly be taught in the field or by handling actual objects. Videodiscs can provide a comprehensive collection of visual information: a single videodisc can hold over 100,000 high-quality images, each one being directly accessible with an access time of at most a few seconds. In addition to still frames, collections of frames forming moving sequences may also be stored. The still frames or moving sequences may then be displayed on a television screen, or on a computer screen overlaid with standard computer-generated text or graphics.

When controlled by a microcomputer program and linked up to a conventional database, interactive videodisc technology provides very powerful and flexible methods of managing and retrieving visual, and related textual and numerical, information. One could envisage, for example, artefact numbers and locations from a database being superimposed on a full colour photograph of an area under excavation: an individual find could then be selected from the screen and displayed, possibly as seen from any of several different angles, along with any associated notes.

Interactive video has a huge potential for training purposes which has been recognised for some time in industry and is gradually becoming exploited in secondary and higher education, despite the high cost of actually mastering a videodisc. The technology is ideal for independent student use or, with networked workstations sharing a single resource, for tutorial group teaching.

* Department of Computing Studies,
University of Leicester,
University,
Leicester LE1 7RH

For a background on the development of interactive videodisc technology the reader is referred to Martlew 1988.

37.3 The software components of interactive videodisc systems

Three software components can be identified in an interactive videodisc system.

The first and most obvious is an indexed collection of images. The images are stored in analogue form on the videodisc as videodisc *frames*, indexed by a unique frame number identifier.

The second component is a standard database of textual or numerical information. This may also be held on the videodisc, but can be held separately. The database must be linked to the collection of images: thus *FrameID* is a record type which will appear in relationships forming part of the database structure (see Richards & Ryan 1985, p. 165).

The final component is the control software which enables a user to explore and retrieve information in the database and to display selected images.

37.4 The use of interactive video in training: teaching methodologies

Many systems used for industrial management training comprise a videodisc consisting mainly if not exclusively of moving sequences, together with a program which takes the user (a student) on a learning path through them. At various points the student is asked a question; future progress depends on the response. While different students will progress in different ways, their progress will depend only upon their responses and the wishes of the original programmer.

This rigidly-structured approach reflects the sort of methodology prevalent in many early 'computer-aided learning' (CAL) programmes in disciplines where an accumulation of factual information is especially important, such as law and the medical sciences.

The philosophy of the well-known BBC Domesday system is very different. Here, the emphasis is upon allowing the user total freedom to investigate and interact with the visual information and the associated database. There are no questions to be answered; instead a student learns by exploring, although this might well be done under tutorial guidance and certain goals might be set by the tutor.

A third approach is provide facilities which allow a tutor to construct his own 'tutorials'. These so-called 'authoring' facilities are designed to enable tutors, who are not necessarily computer specialists themselves, to construct rigidly-structured tutorials suited to their own needs from a large databank of visual material. The tutor is allowed free-exploration facilities, and uses them to select sequences of videodisc frames, interspersing them with his own comments and points where a response is required from the student (e.g. to a multiple-choice question) and using some simple flow-of-control structures to construct different pathways depending on the response.

37.5 Evaluation

In the rigidly-structured approach student responses may be directly monitored and students 'marked' on the basis of these. This sort of approach may be favoured where CAL exercises are used as part of a continual assessment assignment.

There is no mechanism for 'marking' a student in such a direct way in the free exploration approach. It is of course possible to monitor a student's use of the system and a tutor might wish to use this as the basis for assessment. However it seems unlikely that a tutor favouring this sort of teaching methodology will be unduly concerned with the step-by-step assessment of individual students. More probably, such monitoring will be used as a class statistic, enabling the tutor to assess the overall success or otherwise of the system.

37.6 Some problems with existing systems

In many existing interactive videodisc systems, the control software is specific to a videodisc. This is scarcely surprising in the case of many of the systems used in industry, since they are produced in order to teach a management trainee about a specific problem. The same is in fact true of the BBC Domesday system; both the database and the control software are specific to the system. In fact, both are encoded onto the two videodiscs, and are not obtainable separately.

This approach is understandable in the case of a commercially marketable videodisc system where financial concerns are paramount and software piracy a major concern. It does however place strong restrictions on portability and updateability. Videodisc mastering is expensive (although once it is done, obtaining copies of the videodisc is relatively cheap). If the associated database and/or control software are encoded onto the videodisc, they can not be updated or amended without a new disc being mastered at great expense.

'Authoring' programs, and videodisc systems whose software includes authoring facilities, have recently started to appear. However they are often highly specific to a particular subject area, user-unfriendly, and tied to particular hardware configurations.

37.7 Objectives of the Leicester project

The most immediate objective of the Leicester project is to produce a working pilot system so that its use for teaching undergraduate archaeology can be evaluated in the short term.

Our longer-term objective is to produce powerful and generally applicable software for the use of collections of images held on interactive videodiscs in higher education and training.

Since this software is intended for the public domain it is essential to make it available independently of the collection of images on a particular videodisc. The same must be true of any database to be used in association with a particular image collection. Different databases might then be used with a given collection of images, or different collections of images with a given database; the control software could be used with a number of different videodiscs; and updated versions of the databases or control software could be distributed without causing any difficulties.

37.8 Basic software requirements

Our control software must be expandable and portable to different hardware configurations. It must also be reasonably 'future-proof': while the storage and retrieval of

high-quality images is a stable development, interactive videodisc technology is merely a particular way of achieving this and will before long be overtaken by alternatives.

We wish to minimise the 'user familiarisation' necessary—both for students and tutors.

We wish to ensure 'academic portability': that is to ensure that the software will not be tied to teaching methodologies prevalent in a particular subject area. The needs of different teachers might fall anywhere between the two extremes mentioned above, *i.e.*

- rigidly structured computer-based tutorials, as favoured in many 'computer-based teaching' programmes in the 'hard' sciences, and particularly in law and the medical sciences; and
- the opportunity for free, or only loosely constrained, exploration of a large database of visual, textual and numerical information, as is epitomised by the BBC Domesday system and generally favoured in the social sciences and humanities.

We also wish to include facilities to record the progress of students, both for the purposes of assessing individual students and for evaluating the general success or otherwise of the system as a whole, or of particular strategies adopted by a tutor, in different teaching contexts.

37.9 Deliverables

The deliverables from the Leicester project will be a mastered pilot videodisc, the associated database, and the control software. Copies of the disc will be available at cost price, and the database and control software free, within British universities.

The pilot system is intended to provide a manageably-sized demonstration of the potential of interactive videodiscs in teaching archaeology as well as a working resource relevant to the needs of undergraduate Archaeology students. The subject area chosen for the pilot disc is prehistoric British ritual sites.

The first version of the pilot system will run on RM-Nimbus PC1 workstations with Phillips videodisc players. Subsequent versions will run on IBM-compatible machines using other players.

In order to make use of the database features within the first version of the pilot system users will need to possess the dBaseIII+ package. Subsequent versions will offer alternatives.

In order to make use of the 'longer response' capability (see Appendix, Section A.6.4) users will need to possess a suitable word-processing package such as Microsoft *Word* or *WordStar*.

37.10 Software development

The author and two Computer Officers within the Department of Computing Studies at Leicester are involved in the software development for the project. In addition, a group of second-year Computing Studies students have been set the design and development of part of the software (a part which it is not essential to have in completed form by mid-1988) as a group project.

Software development methodologies taught to Computing Studies students have been used throughout, in particular the progression from requirements to specification,

design, implementation, and finally building and testing. Requirements documents were produced within the Department of Archaeology and the Specification document, progressing from an overall level to more detailed levels, prepared in response within the Department of Computing Studies. At each stage the Specification has been extensively discussed within the project's Management and Steering committees. Part of the system (the 'Browse Videodisc' option—see below) has been specified using the Formal Development Technique known as the Vienna Development Method (VDM). Design work has proceeded apace and the first skeleton implementation is already completed.

The choice of hardware (networked RM-Nimbus PC1 workstations), of design methodology, and of programming language for software development work (Modula-2) reflect the ability to take advantage of experience and existing program libraries in the Department of Computing Studies.

Database facilities within our package would ideally be fully relational, but since the user must possess a suitable 'off-the-shelf' database package we must (initially at least) choose a widely used and generally available package. For these reasons the database management part of the pilot system will be programmed using dBaseIII+.

There will be a facility to exit from the system to a standard Word Processing package for 'longer responses' (see Appendix, Section A.6.4).

37.11 Description of the software

Part of the high-level specification is reproduced in the Appendix. It consists of a description of the structure of the system (Sections A.1 to A.7) and an overall description of the top-level options (Section A.8). There are nine of these options; one (*Browse Videodisc*—the one which enables a student or tutor freely to browse the videodisc contents) is described in more detail in Section A.9.

37.12 Special features of the Leicester system

Although the Leicester system is described in some detail in the Appendix, we wish to draw attention here to two areas where our software aims to make significant advances over existing systems.

37.13 Flexible authoring facilities

Existing authoring tools, as described above in Section 37.4, simply allow rigidly-structured tutorials to be constructed by different tutors from a large databank of visual material. Students running a particular tutorial are still subjected to a strict CAL methodology.

In order to ensure academic portability (see Section 37.8 above) we wish to cater for teaching methodologies between the extremes of strict CAL and completely unstructured exploration. For example, an archaeology tutor might wish to show the student some particular examples of prehistoric ritual sites in order to make a particular point about their architecture or design, and then to allow the student to explore the database for similar sites, viewing them or 'walking about' them as much as desired, in order to answer a more general question.

Suitably flexible authoring tools are contained in the 'Make Tutorial' option (see the Appendix, Sections A.6 and A.8). Rigidly structured tutorials can be constructed by only displaying particular videodisc frames for information purposes, and only eliciting choice, screen position or single line responses. More loosely-constrained teaching programs can be constructed by making extensive use of free browse options for the student within tutorials, and by eliciting longer text responses.

37.14 The abstract videodisc structure

One of the main features of the Leicester software is the ability to impose an abstract structure upon the indexed collection of images stored on a videodisc. This then gives the user the opportunity to 'browse the videodisc' by moving about this abstract structure, quite independently of the ability to browse the database and access videodisc frames by identifying relevant frame numbers in the database.

A specific example of such an abstract structure is provided by the BBC Domesday system, where the user may browse Ordnance Survey maps, moving up, down, left or right and zooming in or out to different scales.

Our abstract structure is much more general. We have attempted to develop a simply understood but powerful and widely applicable abstract structure for the organisation of an indexed collection of analogue images. The abstract structure is a hierarchy constructed from a few simple abstract concepts. An example of the latter is a *grid*: a collection of frames conceptually organised in a rectangular grid, each frame having up to eight neighbours in the directions up, left, up and left, etc. The edge of a grid may be ragged, but all frames in a grid must be accessible from all others. Grids arise, for example, in the representation of geographical maps, where neighbouring frames contain overlapping maps to the north, south, east and west of the current one.

In normal circumstances an abstract structure would be imposed upon the images on a videodisc (or collection of videodiscs) by the person who mastered the disc, using our special configuration program. To a tutor or student, making subsequent use of the disc, the abstract structure should be natural and need little extra thought, the relationships between frames being suggested by the content of the frames themselves. This is of course the case with the operations for moving to adjacent maps or zooming in or out on the Domesday system.

37.15 Timescales

The first deliverable product will be the videodisc, which will be mastered in May 1988. Copies will be available shortly afterwards.

An intermediate version of the control software will be completed by the summer so that evaluation of the system can commence in the academic year 1988-89. Two workstations have been assembled in a room dedicated to the project in the Archaeology Department at Leicester University. A further workstation is to be set up in the Department of Prehistory and Archaeology at the University of Sheffield. The initial evaluation will take place at both locations.

An advanced version of the software will be available in March 1989, and will be demonstrated at CAA89. The final version of the software will be available not later

than the summer of 1989, and it is hoped that wider usage of the system will take place during the following academic year.

37.16 Conclusion

The Leicester project seeks to solve a significant problem in handling archaeological information for teaching purposes. The solution will be applicable in any discipline where large amounts of illustrative material are used, such as geography, geology and medicine. It will make available, at low cost, ways of using information which were not practicable before the development of videodisc technology.

References

- MARTLEW, R. D. 1988. "Optical disc storage: another can of worms?". in Ruggles, C. L. N. & Rahtz, S. P. Q., (eds.), *Computer and Quantitative Methods in Archaeology 1987*, International Series 393, pp. 265-268. British Archaeological Reports, Oxford.
- RICHARDS, J. D. & N. S. RYAN 1985. *Data Processing in Archaeology*. Cambridge University Press, Cambridge.

The software is written in Pascal and runs on the following hardware: IBM PC compatible, 80286 or 80386 processor, 1 MB RAM, VGA or EGA graphics card, mouse, and hard disk. The software is written in Pascal and runs on the following hardware: IBM PC compatible, 80286 or 80386 processor, 1 MB RAM, VGA or EGA graphics card, mouse, and hard disk.

The software is written in Pascal and runs on the following hardware: IBM PC compatible, 80286 or 80386 processor, 1 MB RAM, VGA or EGA graphics card, mouse, and hard disk. The software is written in Pascal and runs on the following hardware: IBM PC compatible, 80286 or 80386 processor, 1 MB RAM, VGA or EGA graphics card, mouse, and hard disk.

The software is written in Pascal and runs on the following hardware: IBM PC compatible, 80286 or 80386 processor, 1 MB RAM, VGA or EGA graphics card, mouse, and hard disk. The software is written in Pascal and runs on the following hardware: IBM PC compatible, 80286 or 80386 processor, 1 MB RAM, VGA or EGA graphics card, mouse, and hard disk.

The software is written in Pascal and runs on the following hardware: IBM PC compatible, 80286 or 80386 processor, 1 MB RAM, VGA or EGA graphics card, mouse, and hard disk. The software is written in Pascal and runs on the following hardware: IBM PC compatible, 80286 or 80386 processor, 1 MB RAM, VGA or EGA graphics card, mouse, and hard disk.

The software is written in Pascal and runs on the following hardware: IBM PC compatible, 80286 or 80386 processor, 1 MB RAM, VGA or EGA graphics card, mouse, and hard disk. The software is written in Pascal and runs on the following hardware: IBM PC compatible, 80286 or 80386 processor, 1 MB RAM, VGA or EGA graphics card, mouse, and hard disk.

The software is written in Pascal and runs on the following hardware: IBM PC compatible, 80286 or 80386 processor, 1 MB RAM, VGA or EGA graphics card, mouse, and hard disk. The software is written in Pascal and runs on the following hardware: IBM PC compatible, 80286 or 80386 processor, 1 MB RAM, VGA or EGA graphics card, mouse, and hard disk.

27.15 Interceptor

The first deliverable product will be the software, which will be available in May 1988. Copies will be available shortly afterwards.

The immediate output of the project will be provided by the project to the evaluation of the system can commence in the academic year 1988-89. Two workshops have been arranged in a room situated in the project in the Geography Department at Leicester University. A further workshop is to be set up in the Department of Geography and Anthropology at the University of Bradford. The initial evaluation will take place at both locations.

An advanced version of the software will be available in March 1989 and will be distributed at CAI99. The final version of the software will be available not later

A

Leicester Interactive Videodiscs in Education (LIVE) project—draft system specification

Part I: STRUCTURE OF THE SYSTEM

A.1 Overall structure of the system

The software will consist of the following:

1. A database of textual and graphical information.
2. An indexed collection of pictures, hereinafter 'frames'.
3. In-house software to drive the system.
4. An 'off-the-shelf' database management system, dBaseIII+ in the first instance, and an 'off-the-shelf' word processing system such as Microsoft *Word* or *WordStar*.
5. 'Carousel files', 'tutorial files' and 'student response files', which are files created from time to time by tutors and students and are stored locally.

The images will be stored on each copy of the videodisc; other software will be stored on computer media.

A.2 Organisation of the database

1. The database will be used to structure information pertinent to the subject theme and to relate this information to the videodisc images.
2. The database will be set up using an 'off-the-shelf' database management system, dBaseIII+ in the first instance.
3. The database structure will be designed in a relational manner but implemented within the constraints of the database management system being used. dBaseIII+, for example, does not support a fully relational structure but other systems which it is intended to use in later versions do do so.
4. *FrameID*, the index number of a frame on the videodisc, will be a record type within the database.
5. The detailed structure of the database will be described in the User Manual.

A.3 Physical organisation of videodisc frames

1. A videodisc will in general be organised by 'Chapters', each containing frames relating to a distinct topic. However the collection of frames on a topic can, if desired, span different chapters or different sides of a videodisc, or even span more than one videodisc. (See also A.4.2.)

2. Some groups of frames, hereinafter referred to as 'Movies', may be organised in contiguous sequences on the disc. The frames can then be sequentially displayed at the rate of 25 per second in order to produce a moving video film.
3. The frames will also be organised according to an abstract structure described in Section A.4 below. Where possible, the physical organisation of these frames will reflect the abstract structure so as to give optimal access speeds.
4. Other than as in 1 and 2 above, and as may be possible as in 3 above, the physical ordering of frames on a videodisc or videodiscs will be of no consequence.

A.4 Abstract organisation of videodisc frames

A.4.1

An *abstract videodisc* consists of a number of frames linked by a structure whose general form is described below.

1. An abstract videodisc consists of a number of *items*, arranged in a tree structure. An *item* consists of a list of *entries*, and must contain at least one entry. An *entry* is one of the following: a *grid*, a *walkabout*, a *movie*, a *sequence* or a *single frame*. It contains a *startup frame*, which is generally the frame displayed when the entry is first encountered.
2. A *grid* is a collection of frames conceptually organised in a rectangular grid, each frame having up to eight neighbours in the directions up, left, up and left, etc. The edge of a grid may be ragged, but all frames in a grid must be accessible from all others. Grids arise, for example, in the representation of geographical maps, where neighbouring frames contain adjacent or overlapping maps to the north, south, east and west of the current one.
3. A *walkabout* is a collection of frames representing views as seen from a set of related positions. They are organised so that neighbouring frames represent the views obtained by turning or moving to the left or right, or by moving forwards or backwards. Thus each frame has up to six neighbours. The angle turned, or the distance moved, is flexible and may vary from frame to frame. If sufficient neighbouring frames exist, it will be possible to complete a panorama of views from a fixed spot by repeatedly turning to the left or right. A map or plan may be available in order to show the current position and the possibilities for movement.
4. A *movie* is a sequence of frames which can be run through to form a moving video film. Facilities such as fast wind (picture search) forwards or backwards, and freeze frame and frame stepping forwards and backwards, are also provided. The startup frame of a movie is the *banner frame*, which is displayed until the movie is run. When a movie has finished running, the banner frame is displayed again.
5. A *sequence* is a sequence of still frames which can only be stepped through manually. The startup frame of a sequence is the first frame of the sequence.
6. The startup frame of a single frame is the frame itself.
7. An abstract videodisc contains a unique *top item*. This item has no parent. All other items have one and only one *parent item* and are a *subsidiary item* of their parent. Items may have zero or more subsidiaries.
8. The relationship between a parent and subsidiary item concerns individual frames within the items and may be complex. The following rules, however, apply generally.
 - (a) The relationship concerns only a single entry within the parent item, hereinafter the *parent entry*.
 - (b) One or more frames within the parent entry may be *parent frames*.

- (c) Each parent frame has one or more *pick areas* relating to the subsidiary item in question.
- (d) A *pick area* is an area within a picture delimited by a closed polyline. The pick areas for a given frame are non-intersecting. However when taken together they need not necessarily cover the entire picture.
- (e) The pick areas for a given frame need not necessarily relate to the same subsidiary item.
- (f) A given pick area relates to a particular frame within the subsidiary item concerned.

The *Subsidiary item* operation (see A.9.3) prompts the user to identify a pick area within the current frame. If done successfully, the related frame in the subsidiary item becomes the current frame.

The *Parent item* operation (see A.9.3) selects a frame from the parent item and it becomes the current frame. If the existing current frame is the subsidiary of a frame within the parent item, then that frame is selected. Otherwise the startup frame of the parent entry is displayed.

Example 1 . A map might contain several archaeological sites. Each site might be enclosed within a small pick area. Each pick area would relate to (the startup frame of the first entry in) a subsidiary item which might consist of pictures of the site, walkabouts, or larger-scale plans related together into grids.

Example 2 . An entry might consist of a grid of maps at small scale. A subsidiary item might consist of a single grid of maps at larger-scale. Each frame in the parent entry would comprise pick areas which, when selected, would cause the relevant larger-scale map to be displayed.

The structure described here is specified in VDM metalanguage in a separate, Formal System Specification document.

A.4.2

The abstract videodisc structure may (though in general it will not) link frames in different chapters or on different sides of an actual videodisc, or even link frames on separate videodiscs. (See also 1.)

A.4.3

It is likely that an abstract structure relating to a collection of images will be configured by the person who compiled the collection for mastering. Configuration will be achieved using the LIVE configuration program. Note however that no abstract structure is unique; existing structures may subsequently be extended or edited using the configuration program, and different abstract structures might be constructed relating to the same collection of images—or to different collections with some images in common.

A.4.4

All entries have a short caption which must be provided by the configurator. This is for identification purposes, e.g. when setting up a carousel or tutorial.

Still frames, including those within grids, walkabouts and sequences, may also have a default caption, again provided by the configurator, for display with the videodisc frame.

A.5 Structure of a carousel

A.5.1

The carousel feature provides support for the notion of a tutorial (see Section A.6 below) but does not itself constitute a means whereby a tutor and student can interact.

A.5.2

A carousel is a sequence of videodisc frames. Once constructed, it may be stored for subsequent viewing by other users.

A.5.3

Each frame within a carousel may be captioned or uncaptioned. A caption may be either the default caption (see Section A.4.4) or an alternative specified when the carousel is constructed. It may either be displayed in the default position on the screen or elsewhere as specified when the carousel is constructed.

A.5.4

Information about a carousel is stored in a carousel file which, when loaded by the user within the system, enables a carousel to be viewed or edited.

A.6 Structure of a tutorial

A.6.1

A tutorial provides interaction between a tutor and student in a structured manner. The degree of constraint upon the student may be determined by the tutor. One extreme would constitute a rigid computer-aided learning (CAL) facility. The other would allow unrestricted free browse and response without structuring the student's progress in any way.

A.6.2

A tutorial consists of videodisc frames, text information and prompts, and mechanisms for obtaining student input in various forms. Many paths through a tutorial will in general be possible.

A.6.3

A tutorial may be viewed as a network of nodes, where interaction takes place, connected by paths. The interaction at a node may consist of the collection of information by a student and/or their response to a particular problem. When a response has been elicited, the path followed after leaving the node may depend upon the response. The level of constraint in the process of information collection and response is determined by the tutor.

A.6.4

One of four forms of student response may be elicited at a node:

1. A choice (*i.e.* the selection of one of a number of options offered, as in a multiple-choice question). The subsequent path will in general depend upon the option chosen.
2. A screen position. The subsequent path will in general depend upon which, if any, of a number of 'pick areas' the given position lies within.

3. A single line (up to 80 characters) of text. The subsequent path will in general depend upon the presence or absence of certain character strings within the line of text.
4. A longer piece of text. There is no choice of subsequent path.

Student responses of types 1–3 will automatically be logged by the system and stored in a 'student response log file' (see Section A.7 below). Responses of type 4 will be obtained by allowing the student to use a suitable word-processor and will be stored in a 'student response document file'.

A.6.5

Information about a tutorial is stored in a tutorial file which, when loaded by the user within the system, enables a tutorial to be run or edited. Safeguards are incorporated in order to prevent misuse of the system by students.

A.6.6

A student may 'save' their current position within a tutorial and resume later.

A.7 Structure of a student response file

A 'student response log file' contains:

1. A report on responses of types 1–3 (see A.6.4 above). This is in a form which facilitates marking by the tutor.
2. A report on the student's path through the tutorial. This is in a form which facilitates analysis for the purposes of evaluation.

Part II: OPERATIONS

A.8 Top-level operations

A.8.1

The following nine options will be available upon entry to the system: *Browse videodisc*, *Browse database*, *Make carousel*, *View carousel*, *Make tutorial*, *Run tutorial*, *Print*, *Help* and *Quit*. These operations will be invoked from a display screen by clicking the mouse in suitable response areas.

1. *Browse videodisc* enables a student or tutor to browse the videodisc freely. Default captions (see Section A.4.4) are displayed wherever they are provided.
2. *Browse database* enables a student or tutor to search the database freely. Individual videodisc frames may be displayed during the course of such a search. Also, lists of frame numbers and their captions may be saved to files for future use in carousels or tutorials.
3. *Make carousel* enables a user to prepare a carousel, which is sequence of stills captioned or not as required, by one of the following methods:
 - constructing it from scratch;
 - constructing it from a list of frames generated by a database query or queries; or
 - editing an existing carousel.
4. *View carousel* enables a user to view a prepared carousel.

5. *Make tutorial* enables a tutor to prepare a tutorial, either by constructing it from scratch or by editing an existing one. Carousels may provide the basis for information collection within a tutorial (see A.6.3 above).
6. *Run tutorial* enables a student to work through a prepared tutorial.
7. *Print* enables a user to obtain a listing of the contents of a carousel or tutorial, either on a printer or directed to a text file.
8. *Help* gives access to information pertinent to operations 1-7 and their various sub-operations, structured accordingly.
9. *Quit* exits the system and returns control to the operating system.

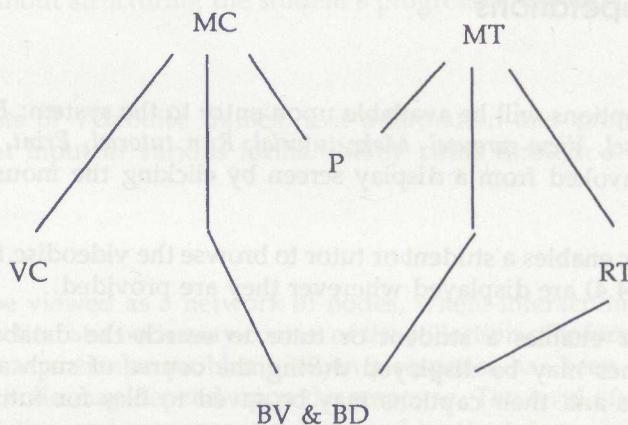
A.8.2

Within these options, drop-down menus will give access to various sub-options. All options will also be available by using the keyboard, and they will often in addition be accessible by using the screen cursor and mouse. The left-hand drop-down menu will give access to other top-level operations available from within the current one (see A.8.3 below), together with *Exit to WP* (exit to word processor) when it is available (see A.8.4 below), *Help* (see A.8.5 below) and *Quit*, which will return the user to the operation from which the current one was called.

A.8.3

Some of the top-level operations are available from within others. Thus, *Browse videodisc* and *Browse database* may be called from within *Make carousel*, *Make tutorial* or *Run tutorial*; *View carousel* may be called from within *Make carousel*; *Run tutorial* may be called from within *Make tutorial*; and *Print* may be called from within *Make carousel* or *Make tutorial*.

The available structure is illustrated by the following:



A.8.4

Exit to WP is an option which enables the user to exit from the system and edit a document using a word processor. When the user quits from the word processor, action is resumed within the system from the point of departure. *Exit to WP* is available within *Browse videodisc*, *Make carousel* and *Make tutorial*. It may also be inserted within a tutorial when a 'longer text' response (see A.6.4 above) is required. However a tutor must appreciate that the use of this facility results in a loss of security, it being possible for a student to access the filestore and operating system from within many word processors.

A.8.5

Help is a generic operation, available to a user at any time. When invoked other than from the top-level menu, however, only the information relevant to the current situation will be available.

A.8.6

Only *Browse database* involves access to the database (with the exception of other operations when they invoke *Browse database*). *Make carousel* and *Make tutorial* will invoke an in-house screen editor.

A.9 Browse videodisc

A.9.1

The operations within this option may be invoked in three different ways:

1. using drop-down menus;
2. using keyboard characters; and
3. using 'mouse shortcuts' (see A.9.5 below).

All available operations may be invoked at any time by methods 1 or 2; most available operations may also be invoked at most times by method 3.

A.9.2

The operations allow the user to display frames by moving about the various abstract structures described in Section A.4 of Part I. The available options at any time are indicated by those drop-down menu options which are highlighted. Initially, the startup frame of the first entry in the top item is shown.

A.9.3 Drop-down menu selection

A.9.3.1

The menu bar will contain the following menu headers:

LIVE Tree Item {Entry} Options Measure

where {Entry} is **Grid**, **Walkabout**, **Movie**, **Sequence** or **SingleFrame** depending upon the nature of the current entry.

The drop-down options within the various menus are as follows:

LIVE	Tree	Item
[Browse Videodisc]	Top item	First entry
[Browse Database]	Parent item	Next entry
[Make Carousel]	Subsidiary item	Previous entry
[View Carousel]		
[Make Tutorial]		
[Run Tutorial]		
[Print]		
Exit to WP		
Help		
Quit		

Grid	Walkabout	Movie	Sequence
Startup frame	Startup frame	Banner frame	First frame
Move left	Move forwards	Run from start	Next frame
Move right	Move backwards	Run from here	Previous frame
Move up	Turn left	Fast forwards	
Move down	Turn right	Fast backwards	
Move up & left	Move left	Freeze	
Move up & right	Move right	Next frame	
Move down & left	Move left	Previous frame	
Move down & right	Move right		

Options	Measure
Hide menus	Define units
Hide captions/Show captions	Define scales
Mouse shortcuts off/Mouse shortcuts on	Measure area
Mouse shortcuts priority/Pick area priority	Measure line
Hide prompts/Show prompts	

Square brackets indicate permanently unavailable options. '/' indicates toggling options. There is no drop-down menu corresponding to the **SingleFrame** header.

A.9.3.2

Selecting a menu item from the *Tree*, *Item*, *Grid*, *Walkabout* or *Sequence* menus will simply cause a new videodisc frame to be shown, except in the case of the following:

1. *Subsidiary item*. A message 'select relevant area or click both mouse buttons to abandon' will be displayed. Note that the relevant pick areas (see Part II, Section A.4.1, 8 above) may or may not be marked on the videodisc frame. The relevant subsidiary item will be selected by clicking either mouse button with the cursor within a pick area. The operation is terminated if the mouse is double-clicked, regardless of the cursor position.

The relevant subsidiary item may also be selected simply by clicking either mouse button with the cursor within a pick area, thus short-cutting the drop-down menu, depending on the option settings (see A.9.6 below). When the cursor is within an 'active' pick area, it changes to a 'Pick area' icon.

2. *Show map*. The walkabout frame will be replaced by a map, which might either be a videodisc frame itself, overlaid by computer graphics, or entirely graphics-generated. The current position and orientation will be indicated. A box labelled 'resume' will be displayed. When either mouse button is clicked with the cursor within the box the interrupted walkabout frame will be re-displayed.

A.9.3.3

Selecting a menu item from the *Movie* menu will either cause a moving frame sequence or a still frame to be displayed (see Section A.4.1, 4).

A.9.3.4

Selecting a menu item from the *Options* menu will have the following effects:

1. *Hide menus*. The drop-down menus will disappear. A small area will become available which, when the cursor is positioned within it and either mouse button clicked, will restore the drop-down menus.

2. *Hide captions/Show captions.* These determine whether the default caption, wherever one is provided (see Section A.4.4), is shown when a videodisc frame is displayed. The default is *captions shown*.
3. *Mouse shortcuts off/Mouse shortcuts on.* These disable/enable the 'mouse shortcuts' listed in A.9.5 below. The default is *mouse shortcuts on*.
4. *Mouse shortcuts priority/Pick area priority.* This sets the priority when a frame with pick areas is being displayed (see A.9.6 below). The default is *pick area priority*.
5. *Hide prompts/Show prompts.* These determine whether the prompts generated by operations such as *Subsidiary item* and *Measure area* are shown. The default is *prompts shown*.

A.9.3.5

Selecting a menu item from the *Measure* menu will have the following effects:

1. *Define units.* The user is prompted to name the measuring units being used (the default is 'units'). Common units are recognised and their relationships to one another are known—thus automatic conversions are possible.
2. *Define scales.* The user is prompted to draw a box on the screen. This is done by clicking a mouse button with the cursor at the position of one corner and dragging the mouse to the opposite corner, where the button is released. If the left button is used, any rectangular box may be produced. If the right button is used, the box is constrained to be square. The user is then prompted to indicate the length of the horizontal and vertical sides of the box in the current units.
3. *Measure line.* This option is only available if scales have been defined. The user is prompted to indicate a sequence of points using the mouse cursor. The left button is used to indicate all points but the last; the right button is used to indicate the last point. Straight lines are drawn joining each point to the next. The distance along the polyline is displayed.
4. *Measure area.* This option is only available if scales have been defined. The user is prompted to indicate a sequence of points using the mouse cursor as above. Straight lines are drawn joining each point to the next and the last point to the first. The interior area of the region demarcated in this way is displayed.

A.9.3.6

Selecting *Exit to WP, Help* or *Quit* from the LIVE menu will have the effects described in A.8.4, A.8.5 and A.8.2 respectively.

A.9.4 Keyboard selection

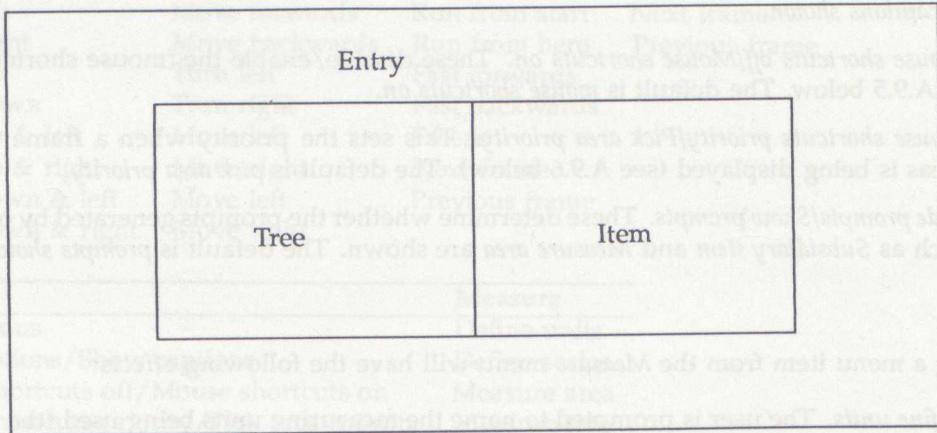
All operations will also be invokable by using sequences of keyboard characters following the escape character. When the escape key is pressed prompts will appear listing the available options. For example, 'ESC T T' will invoke 'Top item' under 'Tree'. 'ESC ?' will invoke *Help*.

A.9.5 'Mouse shortcuts' selection

A.9.5.1

Most operations will also be invokable by clicking the mouse with the cursor in suitable 'response areas' on the display screen. This feature is intended for the user who has become familiar with the system through the use of menu operations and now wishes to short-cut them wherever possible.

The general layout of the response areas is shown below.



The relative proportions of these areas are not specified precisely. They will be identifiable by the user by the fact that the mouse cursor will change form as it passes from one area to another.

A.9.5.2

When moved into the *Tree* area the cursor will change to a 'Tree' icon. The mouse options will then be as follows:

- *Parent item.* Left click.
- *Subsidiary item.* Right click. (Note that this results in the display of a message—see A.9.3.2, 1 above.)
- *Top item.* Double click.

If an entry is unavailable, no action will result when the relevant mouse button is clicked.

A.9.5.3

When moved into the *Item* area the cursor will change to an 'Item' icon. The mouse options will then be as follows:

- *Previous entry.* Left click.
- *Next entry.* Right click.
- *First entry.* Double click.

If an entry is unavailable, no action will result when the relevant mouse button is clicked.

A.9.5.4

The form of the mouse cursor in the *Entry* area will depend upon the nature of the current entry (grid, walkabout etc.) and upon the screen position. For example, if the cursor is moved to the right-hand side of the screen when the current entry is a grid, it will change to a right-facing arrow icon indicating that if the mouse button is clicked in this position *Move right* will result.

Unavailable options (such as *Move right* in a grid, when no right move is possible from the current frame) will be indicated by the cursor changing to a 'no movement possible' icon.

- *Grids*. Adjacent frames in grids may be invoked by pressing either mouse button towards the appropriate edge or corner of the screen. A double click in any position invokes *Startup frame*.
- *Walkabouts*. The mouse/cursor options are as follows:
 - *Startup frame*. Double click in any direction.
 - *Turn left/right*. Left click towards left/right of screen.
 - *Move left/right*. Right click towards left/right of screen.
 - *Move forwards/backwards*. Left click towards top/bottom of screen.
 - *Show map*. Right click towards top or bottom of screen.
- *Movies*. The mouse/cursor options are as follows:
 - *Banner frame*. Double click towards top or bottom of screen.
 - *Run from start*. Left click towards top or bottom of screen.
 - *Run from here*. Right click towards top or bottom of screen.
 - *Freeze*. Double click towards left or right of screen.
 - *Fast forwards*. Right click towards left of screen.
 - *Fast backwards*. Left click towards left of screen.
 - *Next frame*. Right click towards right of screen.
 - *Previous frame*. Left click towards right of screen.
- *Sequences*. The mouse/cursor options are as follows:
 - *First frame*. Double click anywhere in area.
 - *Next frame*. Click either mouse button towards right or top of screen.
 - *Previous frame*. Click either mouse button towards left or bottom of screen.

If an entry is unavailable, no action will result when the relevant mouse button is clicked.

A.9.6

When the current frame has pick areas, there is a potential clash between the ways of invoking operations using the cursor and mouse 'shortcuts' listed in A.9.5 above, and the way of calling up subsidiary frames directly using the pick areas (see A.9.3.2, 1).

The toggling options *Mouse shortcuts on*/*Mouse shortcuts off* and *Mouse shortcuts priority*/*Pick area priority* allow this clash to be resolved in a way suitable to the user.

If mouse shortcuts are disabled, then no clash arises. If, however, mouse shortcuts are enabled, as is the default, then the situation is as follows.

When pick area priority is set (see A.9.3.4, 4 above), as is the default, the pick areas associated with the frame override the 'shortcut' response areas. Areas outside pick areas continue to function as response areas. However if mouse shortcuts priority is set, the pick areas may only be selected by explicitly invoking *Select subsidiary item* using the drop-down menu, mouse shortcuts (see A.9.5.2) or keyboard.

The current situation will be obvious to the user because of the cursor icon (see A.9.3.2, 1 and A.9.5).

A.9.7 Example

A.9.7.1 Configuration

Suppose that a videodisc contains map coverage of Great Britain at three scales, from one where the entire country fits into two frames down to 1:50000 (as per the Domesday system). It also contains pictorial information about various archaeological sites. The abstract structure might be configured as follows.

The top item consists of a single entry: a grid containing the two smallest-scale maps. The startup frame is the southern map.

The top item has a single subsidiary item. This also contains a single entry: a grid containing the intermediate-scale maps.

The latter item again has a single subsidiary item. This single 'third-level' item again contains a single entry: a grid containing complete 1:50000 map coverage of Great Britain.

The third-level item has many subsidiary items at the fourth (and lowest) level. Each fourth-level item contains pictorial coverage of a particular site.

An item on, say, Stonehenge might contain four entries: an aerial photograph of the site (single frame), a sequence of plans showing its phases of construction and use (sequence), a walkabout of the site (walkabout) and a film of midsummer sunrise (movie).

A.9.7.2 Use

When *Browse Videodisc* is invoked the startup frame of the top item is displayed, *i.e.* the map of the southern half of Great Britain.

The Stonehenge item might then be selected by three successive 'Subsidiary item' operations. In each case the relevant pick area is selected using the mouse in order to 'home in' on the site. The technique is similar to that used in the Domesday system.

When the Stonehenge item is first selected the startup frame of the first entry will be displayed. This is the aerial photograph of the site. At this point the user has two options:

- to exit from the Stonehenge item, either back to the 1:50000 map from which it was selected, by invoking *Parent item* on the *Tree* menu, or back to the top-level map, by invoking *Top item*; or
- to progress to the next entry on Stonehenge, by selecting *Next entry* on the *Item* menu.

The next entry on Stonehenge is a frame sequence. When the entry is first selected, the first plan in the sequence will be displayed. The user may now move forwards or backwards at will through the sequence by invoking *Next frame* or *Previous frame* on the *Sequence* menu. At any point in this process they may exit from the sequence, either by moving to the next or previous entry on Stonehenge or by exiting the Stonehenge item altogether and going back to the maps.

If the sequence of Stonehenge plans is returned to at a later stage, the first frame in the sequence (the startup frame) will again be displayed initially.