GIST: AN OBJECT-ORIENTED APPROACH TO A GEOGRAPHICAL INFORMATION SYSTEM TUTOR

J.F.Raper and N.P.A. Green
Department of Geography, Birkbeck College,
7-15 Gresse Street, London W1P 1PA

ABSTRACT

Experience gained in the construction of the world's first GIS tutor, ARCDEMO (Green 1987), has emphasised the importance of accommodating different student learning strategies. ARCDEMO, while highly successful (it has been accessed over 2000 times via JANET-the UK Joint Academic Network) suffered from static graphical displays, a single predetermined access path, and an overall design which made alteration and updating of its material problematic. Object-oriented programming languages offered a means of addressing these problems and were also attractive in respect of the low development resources required. The Geographical Information System Tutor (GIST) exploits this approach using Apple's HyperCard software and incorporates a "point-and-click" interface with graphical cues to initiate operations which include animated demonstrations, step-by-step illustrative graphics and graphical displays capable of user modification. The topics covered by GIST, when taken together, define a set of core activities within GIS which can be used as the basis for a curriculum.

INTRODUCTION

The study of Geographical Information Systems (GIS) is a new and rapidly growing field in geography. These systems, although large and powerful, are difficult to use and require a good grounding in spatial theory to exploit their full potential. While computer-aided learning (CAL) techniques offer the ideal means of satisfying the rapidly growing demand for instruction in the use of these systems (Alessi and Trollip 1985), traditional approaches to tutor construction appear to be of little use in meeting the special requirements of an application based on complex graphical manipulation (Rhind 1988).

THE ARCDEMO PROJECT: A PERSPECTIVE

The ARCDEMO project began as a response to the wide interest in the instalment of ARC/INFO at Birkbeck College, University of London, one of the first geographical information systems in Europe. Since the purchase was partly funded by the UK Economic and Social Research Council (ESRC) an early attempt in 1984 was made to make a demonstration and tutor available by interactive logon over JANET. Subsequently, ARCDEMO was installed at a number of sites in the UK and abroad.

Development of ARCDEMO

ARCDEMO was written in both DEC VAX command language (DCL) and FORTRAN, and runs under the VMS operating system (Green 1987). The system produced graphic output for Tektronix (or compatible) graphics terminals and used plot files to generate the graphics used in the displays. The main subjects covered included map editing, network analysis, projection change, buffering and map overlay. The names and affiliations of those who accessed the system were stored, and mail could be sent by users directly to the system manager.

Extending and improving ARCDEMO

ARCDEMO suffered from a number of shortcomings, most of them system dependent. Thus, the use of ARCDEMO on a heavily used multi-user machine meant that the system suffered from variable response rate. Also since ARCDEMO was mostly used over a network with a restricted transfer rates and variable reliability, access to the system could be a problem.

In addition, ARCDEMO although a menu driven system suffered from a relatively inflexible structure. For example, although the various sections of the demonstrator could be read in any order, it was not possible to jump into or out of the middle of a section. The sequence of text and graphics could also only be read forwards, and to re-read a single page the whole section had to be read again. Similarly, graphics are stored a single images and cannot be merged or overlaid, and since the screen is cleared between each graphic no visual effects can be used, for example, to dissolve images into each other. Thus, the graphics act simply as colour illustrations.

ARCDEMO was also limited by the speed with which graphics could be generated and displayed. This made it necessary to compress a number of different concepts onto just a few images, sometimes relying on fairly crude graphics to convey these ideas. For example, the section in ARCDEMO which deals with "cleaning" vector data introduces a large number of subjects and terms including map registration and transformation, generalisation by coordinate thinning, node snapping and overshoot error correction in just 3 images.

THE DESIGN AND CONSTRUCTION OF GIST

Whilst the ARCDEMO project met the demand for a system to illustrate the capabilities of GIS in general and ARC/INFO in particular, the growing needs of in-house teaching and the explosion of new GIS software required the design and implementation of a more sophisticated tutor. The project to develop a new tutor was therefore initiated in May 1988.

Design criteria for GIST

The design of tutors for geographical information systems presents some significant challenges to the developers of CAL tools. The key problems identified in this project were the human-computer interface (HCI), the need for interactive graphics and an overall flexibility in the design structure to cater for different learning strategies.

Recent research has demonstrated the value of the use of highly interactive systems in the learning process. Systems using window managers, icons, mice, and pop-up menus (WIMP) interfaces have become popular following their development in the Smalltalk-80 project at XeroxPARC (Goldberg and Robson 1983) and successful implementation on the Apple Macintosh. These techniques now represent a new orthodoxy in the HCI for microcomputers and workstations. Recent research by the Gartner Group (1987) showed how computer users were able to learn more quickly <u>and</u> become fully familiar with more applications when using a consistent WIMP interface, rather than a standard command line system. This can be traced to the use of visual cues in the icons, the ability to select options from menus and the execution of commands using the mouse. The learning overheads on the user can be reduced by making the system easy to use: this leaves more time and attention to be devoted to the subject matter.

A second key problem identified concerned the predominance of graphical subject matter required to teach GIS concepts. To illustrate these concepts requires sophisticated graphical capabilities to display maps and the operations carried out on them. However, to exploit fully the advantages of the graphical display the user should also be able to interact with the maps and images presented. Accordingly, a WIMP interface was seen as necessary to simulate real graphical operations and to allow the illustration of multi-step procedures such as digitising using animation techniques.

Thirdly, the development of CAL tools to fulfil a wider range of learning strategies was seen as desirable (Watson 1987). Many of the available tools for building tutors require the mastering of a complex authoring language such as PILOT and tend to enforce a 'forward working' mode of learning (Walker and Hess 1984). These systems are also designed to provide answers to specific problems in a well-defined (and limited) domains. This question/ answer mode of operation was considered inappropriate for the design of a GIS tutor where the knowledge domain is large, rules are relatively difficult to define and users often have limited knowledge of the subject area (Green 1986).

Hypermedia systems seemed to offer the best opportunity to address these problems since they stress the <u>linkages</u> between subjects and formats of presentation, for example in text, static graphics, animation and video (Ambron and Hooper 1988). The successful implementation of these concepts has however only recently become possible with the development of object oriented languages (OOL). In 1987 Apple Computer released 'Hypercard' for the Macintosh (Goodman 1987) which was amongst the first OOL for a microcomputer (implementing the language 'Hypertalk'), and which went much of the way to providing the functions of a Hypermedia tool.

Hypercard combines the functions of a flexible database management system with a high level OOL into an integrated system. The software uses the metaphor of a card "<u>stack</u>" to form a simple and easily understood analogy, and includes graphical and text tools for the generation of visual materials. The "cards" in the stack contain text and graphics which may be unique to one card or be shared by all the cards in a stack. Cards are linked using "buttons"- active areas of the screen which when pointed to by a user with the mouse will carry out some action specified by an accompanying program or "script". Text information is stored in "fields" whose font, size and style may be defined; each field can also be made to scroll in order to store large amounts of text on a single card.

In addition each card is made up of a number of layers which contain fields, graphics and buttons. Some layers belong only to a single card whilst others may belong to all or part of a stack. Items stored on the "background" layer can be occluded by foreground items which belong to a single card, and items may be made transparent so for example text or graphics can be viewed through a text field or button. The system is event-driven where the user initiates actions to invoke a script, which may be associated with any of the "objects" discussed. The script is activated upon the receipt of a "message" sent by the event which is passed along a hierarchy of objects by a message "handler" until it reaches the appropriate object. Hypercard can also be extended by adding external commands and functions compiled in the Pascal or C programming languages.

The combination of Hypercard and the Macintosh with its WIMP interface seemed to represent the ideal environment for the development of an interactive GIS tutor. The ability of the system to display and integrate graphics (including animation) was attractive as was the opportunity to establish an "open" tutor with cross-referencing links and powerful search capabilities. Hypercard is also easy to use and understand and offered the possibility to develop a tutor quickly and without the extensive testing required by other software. The main disadvantages of Hypercard are that at present it has limited colour facilities and will only run on the Macintosh, although other less sophisticated Hypertext systems do exist for other makes of computer.

Construction and layout of GIST

Having selected Hypercard as the development tool for the tutor the most significant problem to be overcome was the structuring of the information to be presented. The design of the tutor followed a similar line to ARCDEMO in that individual topics were split into separated sections. There was, however, a wider range of topics which could be cover with a more flexible tutor and so many of the section were split into sub-sections. The range of graphical effects and animation capabilities of HyperCard allowed some new opportunities to give greater visual impact to the tutor. The tutor was designed to run exclusively "mouse-driven" and required no commands to be entered at the keyboard.

One of the main advantages of constructing the tutor within HyperCard was that little formal design was necessary; individual sections of the tutor could be developed independently an then linked together using buttons or programmed links. The only important design decision, which was taken at an early stage in the tutors development, was to split individual sections into separate HyperCard "stacks" controlled by an introductory stack. This increased the ease with which new sections could be added to, edited and reorganised.

On execution the tutor opens an introductory stack which provides access to the tutor, help facilities and other features. The tutor displays a title page moving to a "welcome" page describing the main options offered by the tutor together with credits. These options include: an "introduction"; a "table of contents"; a "map" of the tutor; a GIS trade directory; a bibliography of GIS publications; and an "encyclopedia". The introductory section is provided for new or inexperienced users and contains several cards or pages describing how to move through the tutor; the structure of the sections included; the actions taken by the various buttons; and how to use the "table of contents" and "map" facilities.

The tutor "contents" facility is equivalent to the table of content in a book. A one line description of each of the main tutor sections is provided together with a button linked directly to it. If a section is selected by 'clicking' on the appropriate button, then a title page for that section is displayed which contains a brief description of its contents. Section title pages also usually contain buttons linked to sub-sections (see, for example, figure 1 below) and also other main sections, the contents and the map.

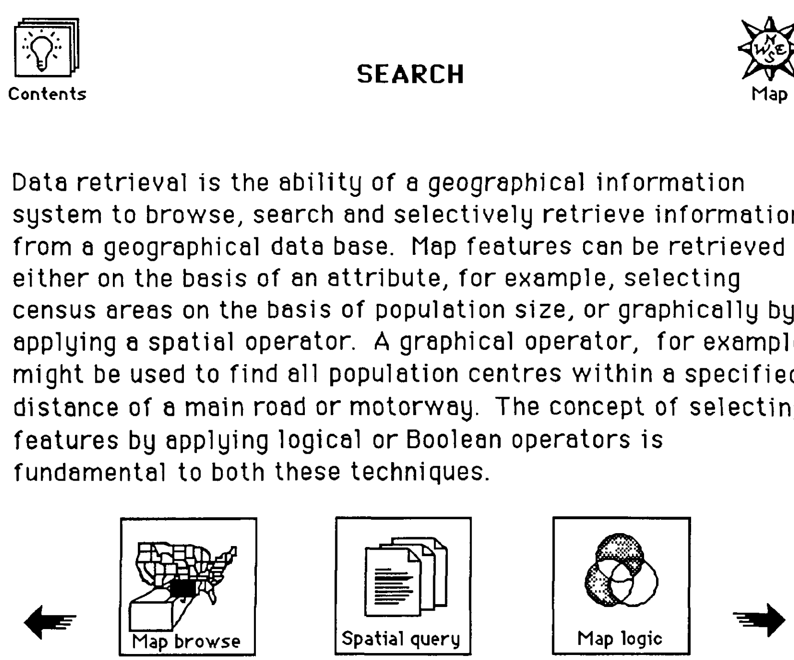

Figure 1 Title page for the "Search" section

The "map" facility (see figure 2 below) provides a visual index to the tutor. The map depicts the structure of the tutor sections and sub-sections and also provides instant access to any of these components. All the sections and sub-sections illustrated in the "map" are also buttons, which provides a browse facility for users with a specific interest. The "map" is accessible from almost any point within the tutor to facilitate this process.

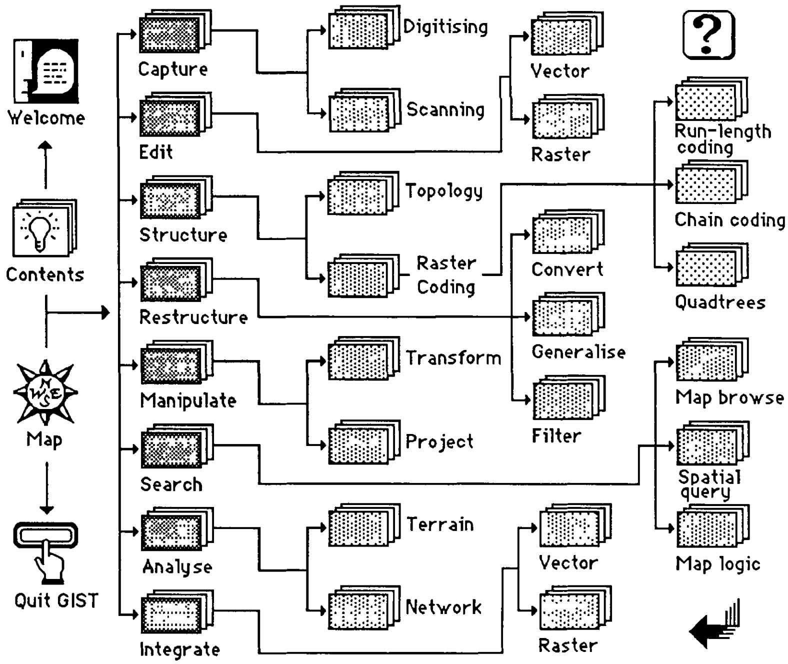

Figure 2 Map of the sections and sub-sections in GIST

The trade directory and bibliography options provided on the "welcome" page are linked to databases which supplement the information given in the tutor. The trade directory provides information on approximately 35 GIS systems currently available in the UK. The address of the company marketing the system is provided together with a brief description of the software including then form of attribute and graphical storage used, and some basic information on hardware requirements. The bibliography lists recent publications which should be both easily available and understandable to those with a limited knowledge of GIS and is arranged by sections covered in GIST. The bibliography also provides details of journals, newsletters, etc. which often carry articles on GIS and related topics (eg. Mapping Awareness, The International Journal of Geographical Information Systems and GIS World) and gives the publishers addresses.

Finally, the encyclopedia is a further feature provided for those users who wish to look up specific terms of interest. Once a particular subject has been accessed the user has the option to return to the encyclopedia or to continue to remain in that section of the tutor.

In a typical session where the user decides to work through the tutor section by section, GIST maintains a record of the sections and sub-sections completed, flagging them to the user by highlighting the buttons on the section header card. This record is maintained both in this mode or if the user chooses to browse through the sections using the map or the encyclopedia. In order to facilitate browsing or following a train of thought, cross reference buttons are provided to link associated subjects, for example line thinning in the sub-section on 'raster editing', and vectorisation in the conversion sub-section. This implementation of cross referencing and the immediate access to the map from almost any point would be difficult in a non object-oriented system.

GIST AS A GIS TEACHING TOOL

Core activities in GIS

Few curicula exist for GIS topics at present: the range of books available as introductory texts is limited and often existing courses have developed around staff expertise or have become out of date quickly as (often monthly) software developments accumulate. Accordingly the authors decided to design their own approach from first principles, based on an assessment of GIS functionality in spatial operations rather in a software dependent fashion. Thus, the coverage of GIST in version 1.0 is based around the general principles of spatial data handling in a computer system. However, the modular design is easily extendible and further objectives of the project include the addition of application modules, for example illustrations of the handling of remote sensed imagery. The modules defined are described in the sections below.

Data capture is divided in sections dealing with vector and raster methods. In each case, an explanation of the concept is followed by a description of the various techniques. Vector data capture, for example, is illustrated by the processes of manual digitising. The process of encoding a map is made more apparent by animating a digitising session showing the digitising cursor moving over a map sheet along with the generation of line and node features. Raster scanning is used to illustrate the concept of raster data capture, and the technique is shown reducing a map to a pixel image.

Editing is also illustrated using examples applicable to raster and vector data structures. Vector data editing uses the output derived from the vector data capture section to show how simple errors associated with manual digitising can be corrected. In this example, overshoots, undershoots and misplaced points are highlighted and then techniques for solving them described by 'clicking' on the error. An explanation of the problem and how to solve it is then provided, and corrected. Raster editing illustrates gap removal, line thinning and stray pixel removal using a similar technique.

The processing of structuring digital map data (storing the data in a form suitable of analysis and easy retrieval) is also subdivided by vector and raster data types. The concept of polygon topology is illustrated by warping a simple figure to show how topology overall shape are unrelated: the connectivity and adjacency of the figure are shown using simple animation. Raster coding techniques shown include run-length coding, chain coding, and the production of a quadtree from a simple region defined on a grid. In this example, as the figure is recursively sub-divided the quadtree is built alongside and the user can move back and forth to examine the process in detail.

A section entitled restructuring covers the concepts lying behind map generalisation and filtering techniques. The Douglas-Peucker coordinate thinning algorithm and the calculation of a low-pass filter, for example, are illustrated using a stepped animation sequences. This section also deals with techniques for spatial data structure conversion including raster-to-vector and vector-to-raster processing.

Map sheet manipulation is divided into sub-sections dealing with map transformation and projections. Transformations-rotation, translation, scale change and warping are illustrated by animating these operations when applied to a simple grid. The fundamentals of map projections are also explained and examples generated by the ARC/INFO system are used to show the effect of changing the projection of a map of the USA.

The concepts of specifying a search and the retrieval of information using a GIS are covered in 3 sub-sections dealing with map windowing operations; Boolean logic for map operations; and spatial query. These sections rely on a large amount of user interaction including question/answer sessions, the construction and application of selection criteria to a database containing selected population statistics for railway stations in SE England.

Network and terrain analysis are chosen as being typical of the types.of spatial analysis performed by a GIS. A step-by-step description of the fundamental algorithms (such as finding the shortest path through a network) is provided along with several animated sequences illustrating the building of terrain models (including a fly-by of Mount St. Helens!) which is becoming an increasingly important area of GIS research (Raper 1989).

The final section, which deals with map integration, helps to illustrate some of the important differences between the raster and vector data models. Some of the topics covered include: the concept of integrating attribute information; the problems of "sliver" errors generated by vector-based overlay; and the concept of applying "masks" to assist raster integration. Finally, the the section culminates with an animated illustration of vector-based map overlay to select a region fulfilling multiple selection criteria.

Using GIST in a short course programme

The topics above when taken together define a set of core activities within GIS which can be used as a short curriculum. The tutor is already in use in the bimonthly Short Course programme at Birkbeck College (funded by the UK Dept. of Trade & Industry PICKUP initiative), where the objective is to provide an introduction to GIS and spatial data handling.

In order to be useful in teaching GIST contains a report generation facility which allows the instructor to check on the progress of the student through the system. This report is also used to track the interests and aptitudes of the students in order to indicate where to improve and develop the coverage. In teaching the short courses the authors have found that use of GIST individually by a small class creates a productive learning environment since the instructor can respond to individual queries as they arise. The students can also customise their own notes by printing out the cards using Hypercard's own report printing facility, and then annotating them as appropriate.

CONCLUSIONS

A new GIS interface?

A strategic objective lying behind the development of GIST is a wider design approach to the improvement of the interface to GIS. In general GIS systems are difficult to use and a significant learning overhead to most systems lies in training associated with the operating system and the command structure. It is considered that the techniques used in GIST may provide the basis for the design of a generic spatial language interface. Since Hypercard can expand commands to remote computers using communications software such as Sequelink, it is anticipated that some of the tutors' demonstrations could be implemented by handshaking with a real GIS and returning the data to GIST. This has provided the blueprint for the development of a spatial language named UGIX and is discussed more fully in Rhind, Raper and Green (1989) (this volume).

Future developments

A number of UK Higher Education Institutions are currently using GIST in their teaching programmes as an introduction to spatial theory and GIS. GIST version 1 represents the core of a GIS tutor which it is hoped will provide a basis for a wider range of Hypermedia materials currently under development, some of which are to be funded under the UK Economic and Social Research Council's Regional Research Laboratory Initiative.

The authors consider that both the underlying principles and implementation of GIST represent a considerable advance on existing learning materials in GIS. However, in this fast moving industry the requirements for training materials are constantly expanding, and a considerable challenge remains to meet these further needs.

ACKNOWLEDGEMENTS

The authors wish to thank Apple Computer (UK) for the loan of a Macintosh II computer with which to undertake this work.

REFERENCES

Alessi,S.M. and Trollip, S.R. 1985, Computer based instruction methods and development, Prentice Hall.

Ambron,S and Hooper,C 1988, Interactive multimedia, Microsoft Press.

Gartner Group 1987, Apple Computer and its Macintosh: product report, available from Apple Computer.

Goldberg,A and Robson,D 1983, Smalltalk-80: the language and its implementation, Addison-Wesley.

Goodman,D 1987, The complete Hypercard Handbook, Bantam Books.

Green,N.P.A. 1986, User/ system interfaces for geographical information systems, Report 4 NERC Remote Sensing Special Topic: The conceptual design of a GIS for the Council. Birkbeck College, University of London.

Green,N.P.A.1987, Teach yourself geographical information systems. The design, creation and use of demonstrators and tutors. International Journal of Geographical Information Systems, Vol.1, pp. 279-290.

Raper,J.F (ed.) 1989, Three dimensional applications in GIS, Taylor and Francis.

Rhind,D.W. 1988, A GIS research agenda. International Journal of Geographical Information Systems, Vol. 2, pp. 23-28

Rhind, Raper and Green 1989, First UNIX then UGIX, (this volume).

Walker,D. and Hess,R. 1984, Instructional software: principles and perspectives for design and end use, Wadsworth Publishing.

Watson,D. 1987, Developing CAL: Computers in the Curriculum, Harper and Row.