

SyGraf — resource based teaching with graphics

David Wheatley

(Department of Archaeology, University of Southampton, Highfield, Southampton SO9 5NH)

2.1 History of the SYASS project

The Southampton York Archaeological Simulation System (SYASS) is a joint project between the Universities of Southampton and York, and is funded by the Computers in Teaching Initiative (CTI). The project aims to produce an excavation simulation system for use in the teaching of excavation strategy to undergraduate students. Use of a computer simulation was felt to be a promising means of effectively introducing students to the types of decisions that field archaeologists have to make as they compromise the abstract ideal of total recovery against the real world, where pressure from developers and superiors is to reduce costs while still answering archaeological questions. If students could be introduced to this type of decision-making process before (not instead) of letting them loose in the field then this was felt to be advantageous. As a secondary aim the project hopes to encourage awareness of computer technology amongst both students and staff.

A number of similar products already exist including Oxfordshire County Council's *Dig and Settle*, CUP's *Unearthing The Past, Digging Deeper into History* by Roger Martlew for English Heritage and the American *Fugawiland* program. Many of these packages are excellent, but all except *Fugawiland* are aimed at schools, and not at universities. *Fugawiland*, while strong in some respects, is a directed exercise at a regional scale with no facilities for alteration of the scenario by teachers.

A pilot project called CEMYSYASS (Rahtz 1988b) was undertaken in by Sebastian Rahtz. This comprised a simulation based on the database program INGRES in which a data set derived from the Protestant Cemetery in Rome was used as the resource. A simple set of routines were provided with which the user could record details of the gravestones at various levels of detail.

The project employed a Research Fellow for eighteen months to investigate and implement further the aims of the project. Approaches to the simulation were discussed (O'Flaherty 1988a, O'Flaherty 1988b).

The SYGRAF program developed from a short research project undertaken as part of the M.Sc. in Scientific Archaeology (Archaeological Computing) at Southampton University. The program was intended as a re-write of the existing SYASS program, but concentrating effort on producing an intuitive and attractive interface (Wheatley 1989).

Since October 1989, the original program has been substantially re-written, extended and de-bugged. Further work on the code was undertaken and a substantially improved version has undergone some testing with undergraduate students at Southampton.

2.2 Program philosophy and 'virtual archaeology'

It is worthwhile explaining, as the SYASS project has striven to do from its inception, that the SYASS product is not intended to replace (or even to diminish) the fieldwork content of the undergraduate curriculum. Instead, the program is intended to better prepare students before and after they have been into the field, for the sort of decisions they will be required to make. Such an approach may in time reduce the time taken to train students outside the classroom, and therefore actually increase the time available for real excavation.

The SYGRAF database does not attempt to represent an archaeological site as it occurs 'in the ground', but instead records subjective facts and geographical information about archaeological entities in a format which will be familiar to archaeologically literate students. The site representation adopted for the program has been compared to a level 2 or 3 archive of a real site, and this is essentially accurate — the site database contains information which, in field archaeology, would be included at the level 2 or 3 stage (the site is 'phased' for example, and the pottery sherds are identified).

The SYGRAF resource cannot therefore be considered to be contributing to a 'virtual archaeology', and is not intended to do so. Such an essentially reductionist approach neglects the singular most important fact of archaeological interpretation, that of the ascription of meaning to archaeological objects and the theory-laden nature of archaeological data (e.g. Shanks & Tilley 1987). To claim superiority, therefore, by the standards of the ultimate empiricist dream of total recording, is to deny that archaeologists theorize and interpret not objects with no meaning, but *theoretical objects*. In database terminology, no entity can exist unless it is within a universe of discourse, and to include an entity in a data schema is to theorize it.

A 'virtual archaeology' aims also to reduce the amount of practical fieldwork undertaken by field archaeologists by reducing the field project to the status of an experiment which can then be undertaken at a hypothetical level. This ignores the cultural nature of archaeological work, and is a clear attempt to gain credibility by recourse to crude scientific values. SYGRAF should be regarded as a tool for analysing and improving the awareness of students to specific aspects of the work undertaken by field archaeologists, and not as an intellectual 'damage limitation exercise'. The SYGRAF product recognises that archaeological fieldwork is as much a cultural activity as an intellectual one. This attitude is, it is to be hoped, reflected in the SYGRAF program.

2.3 Program design and implementation

The general approach to the simulation has been the same as the CEMYSYASS program (Rahtz 1988b), in that it is a resource-based simulation system. 'Resource based' is here defined as a system whose primary constituent is a database (or resource), and where the program (or programs) used to manipulate the resource have a secondary function within the simulation. This philosophy allows teaching to follow a user-driven approach, which can be contrasted with a more program-directed approach. The latter would, I believe, have created a far less flexible result. Projects which have adopted a program-directed philosophy have frequently suffered from the 'Press return to continue' syndrome (Hawkins 1987). A resource based simulation is preferable, then, for the following reasons.

- It provides a better analogy with reality, where there exists a site (a resource) and a number of ways of extracting information from this resource within a given set of limitations. Decisions are made in reality in an active way; the excavator not only chooses which of the available options to take but decides what the options are in the first place. A program-directed simulation would have been far more limited in this respect.
- It is more flexible. Once a data format for representation of an archaeological site has been adopted, other sites can be coded to act as the resource. By using the simulation program on other databases, different situations can be designed for students.

In addition to these aims of the SYASS project, the requirements of the new program were as follows:

- That the graphics should be interesting and attractive, preferably operating in real-time and re-drawing the site as it excavated.
- The interface should be an intuitive, Graphical User Interface using a mouse and buttons approach. It was felt that this type of interface was most likely to encourage non computer-using students.
- That the new program be sufficiently modular to allow for the addition of new pieces of program code (in other words new excavation activities) at a later date.

In order to implement the program, the underlying representation of the site needed some considerable thought. The primary requirements of the site database (resource) were:

1. The site representation should incorporate depth: that is the site should be layered. The number of layers should not be defined in the program but in the database.
2. The underlying representation of the site should include the graphical description of the site and that the operations which the program performed on the database simulated as closely as possible the real activity of excavation.
3. As much information as possible should be part of the site database, not the program. 'Hardwiring' facts about the site into the program was felt to be compromising the resource-based nature of the simulation.

The specifications discussed above have now been implemented for IBM PC and similar computers. The SYGRAF prototype is a CLIPPER application; CLIPPER is a dBase compiler. This makes it possible to build stand-alone applications for DOS which can perform operations on standard dBase tables, by writing in the dBase language. In addition to this, CLIPPER allows user-defined functions and extensions to the dBase language to be written in C and in Assembler. It was this combination of facilities, (and personal preference of the C programming language) which led to the adoption of CLIPPER as the tool for prototyping the SYASS program. The framework of the program, the routines which control the database operations for excavation *etc.*, are all written in CLIPPER. The graphics and mouse routines (in other words almost all I/O operations) have been written in Microsoft C (though many of the graphics routines, particularly those for text handling in graphics mode are *not* taken from the Microsoft C Graphics Library).

2.4 The resource

The resource is a dBase database, consisting of fourteen tables and three indices. These constitute a relational database in which the site is represented in terms of two types of objects, contexts and finds. Any archaeological object which can be approximated in plan view as a polygon is deemed to be a context; while objects which can be sensibly considered to be spot-finds (in other words have their location recorded only as a point) are deemed to be finds. Contexts are typically archaeological entities such as pits, post-holes, ditches, banks, walls and so on while objects like pottery sherds, coins, flints and animal bones would most naturally be considered as finds.

Finds are defined entirely by single entries in a finds table, and can be equated to point entities familiar to programmers and users of Geographic Information Systems (Burrough 1986, Wansleebe 1988). Contexts are recorded in two tables as simple polygons. Vertices of the context polygons are considered to be attributes of a context, a context may have anywhere between 3 and 99 vertices on its polygon. The vertices are stored in the points table, while the contexts table contains a single record for each context. This form of representation of polygons is the simplest, and so the fastest to perform real-time graphics operations on. The disadvantages of such schemes are, however, well discussed (e.g. Burrough 1986) and mainly stem from the fact that polygons in this schema are topologically independent from one another, and so cannot share edges. So-called weird polygons are also possible, although the requirement for the polygons to be closed precludes the possibility of dead-ends. Spatial analysis using such a representation would be virtually impossible.

The finds and contexts tables also contain fields identifying the layer in which the object occurs and the descriptive attributes of the object. To further increase the speed of the program, the relational model was broken in one important regard — the vertex points stored in the points table are stored in the order they must be drawn to produce a polygon, and to recover from disasters an order attribute was included in the table. Without this attribute, or the assumption that

the points are in the correct order, the representation is incomplete as the points could be drawn in any order.

Importantly, the resource specification includes provision for extending the finds and contexts tables, by adding further descriptive fields to the database structure. These allow the descriptions of finds or contexts to be made quite comprehensive, and allow creators of site databases a degree of flexibility. These extra descriptive fields are then accessible to users of the program from the text-based database section of the program.

The concept of depth is incorporated into the model by including layer attributes. Every find and context has the layer from which it originates as a field, and the layers of the site are considered to be overlain, from layer 1 at the top to however many layers there are, up to a nominal maximum of nine at present. During excavation by the program, the user is only allowed to remove one layer at a time, and is only allowed to excavate lower layers once he or she has removed the overlying layers.

As discussed above, this database does not attempt to represent a site in an uninterpreted format, instead it may be equated with a level two or three archive of an archaeological site, which consists of context records, finds records and phase plans. It was convenient to represent the site in this structured, interpreted way because of the very practical, restrictions of processing speed and data access time — it would obviously have been possible to implement a simulation in which phase definitions were left to the excavator by recording the contexts as three-dimensional shapes. This is possible on mainframe or minicomputers, but is not yet feasible within the restrictions of the desktop microcomputers available to most Universities and other similar institutions. The program had to run at an acceptable speed on IBM Model 50s (and other Intel 286 based computers) and this ruled out real-time three dimensional graphics operations of the sort the program does in two dimensions. Recording the site as a series of phased objects has the additional benefit of making the process of creating new sites easier. Sites are often published in a series of phase plans which can be used as the basis of the stored site. This approach was used in the creation of the Winnal Down site for SYGRAF as used at Southampton University. Plans were digitised in a simplified form from the published report (Fasham 1985), and then further information was appended to the finds and context records later.

Thus, instead of a three-dimensional representation of a site, the program deals with different levels, each represented as a two-dimensional phase plan. This two-dimensionality can, to some extent be offset by the addition of descriptions and by the use of other sources of graphics, most notably the laser disk resource (see below), but these things will never alter the essentially two dimensional nature of the representation.

A facility is provided to link pictures stored on laser disk to the program. This is simply implemented by including a 'frame number' field in both the context and finds database tables. If the laser disk is selected (by a command line option when the program is started) the program sends appropriate codes to the laser disk every time the textual description is requested by the user. As such, the laser disk picture frame

can be regarded as an extra tuple of the relational database, and is treated in exactly this way.

2.5 The program

The program provides facilities for two main activities, excavation and observation. It is mostly operated by a mouse and button interface, which allows the user to intuitively point at buttons on the screen in order to initiate actions, and to point at objects on the screen to simply request information. In addition to this, the program has a more traditional text-based database section, which allows simple queries to be made of the subset of data which represents the user's database.

I do not propose to discuss in detail how the program works, though I wish to identify two features which are significant. Firstly, when the program 'excavates' an area of a layer, the program will convert finds in the resource database into find records in the users database. More importantly, however, it will convert context polygons in the resource database into context polygons in the user's database which represent only that area of the context which would be recovered by the trench. This is accomplished by using a clipping algorithm: the Sutherland-Hodgeman clip. Thus if two trenches each cut the same ditch, the program recovers two separate contexts for the user, each with a unique context number. The only two ways the user has of connecting the contexts, are interpolation or further excavation.

Budgetary control is built into the program. When a user excavates, the program calculates the cost of the excavation based on a simple relationship between area excavated, level of excavation and money. There are four levels of recovery, characterised by four tools, JCB, pickaxe, shovel and trowel. Each of these recover different amounts of detail and consequently are more or less expensive per unit area. The user first chooses which recovery level to use, then identifies the area to excavate. The program then recovers a proportion of the finds, according to the chosen recovery level and subtracts an appropriate amount from the budget. The user must weigh up the cost of excavating at the highest recovery level available (with a trowel), for example, against the destructive nature of excavating with a JCB every time a trench is dug.

2.6 Practical concerns

An exercise using the SYGRAF program was introduced into the curriculum of the first year undergraduate course at Southampton University at the start of 1990. There was, unfortunately, insufficient time to design a comprehensive evaluation strategy for the program, and instead an exercise was set, and this was followed up by a discussion of the program and exercise with the students.

Thirty students were involved, each was required to attend two hour-long practical sessions. Because none of the students had used the program before, two site databases were needed during the exercise. One was reserved for the exercise itself, while the other was provided for the students

to practice with. During the first of the two practical sessions students were introduced to the program, and shown how to operate the program and the associated hardware — most notably the mouse and keyboard. The students were then given a week to get used to the program by excavating the 'practice' site, with any of four computers available to them in the department. A number of problems occurred at this stage. All of these problems were unforeseen, although they seem inevitable in retrospect. All also stem from an astounding lack of basic computer literacy amongst first year undergraduates.

1. There were too many switches on the computers. Many sat in front of a working computer, with the monitor switched off, for some time before plucking up the courage to ask someone. Many neglected to switch on the mains supply.
2. It proved necessary, in some cases, to explain that the wire led from the back of the mouse, and that it should be kept perpendicular to the computer to work properly.
3. Using the DOS command line to start the program proved too difficult for roughly half of the students. Although directories were discussed in the practical, many failed to understand the basic concepts of files and directories. This resulted in some students copying exactly what was shown in the handout (this meant that they typed in the DOS command prompt even though it already appeared on the screen). Fortunately a menu program is now installed on all of the computers, and the program was accessible from this.
4. Converting the budget from the arbitrary financial units used in the program into real terms (excavation capacity) was difficult, and had to be explained a number of times.
5. Roughly half the students failed to understand the form-based database query system after the practical, and had to be shown in more detail.

Although these may seem banal they are listed here to illustrate that computer packages such as SYGRAF cannot be used in isolation from a more general programme of computer education. The first lesson of the SYGRAF teaching experience was that introducing a fundamental computing course to the first-year undergraduate curriculum must be a priority.

There were, however, more positive aspects of the experience: most of the students seemed interested and none seemed totally out of their depth. Many quickly found errors in the handout which may have been misleading (the students were not alone in the learning process) and all mastered the basics of the program eventually.

The second practical was an introduction to the exercise itself. The resource database, which comprised three layers of digitised information based loosely on the Bronze age and Iron age sites at Winnall Down (Fasham 1985) was installed on the computer, and the students were asked 'excavate' the site. The exercise itself was designed to minimise the amount of guidance given to students and so compel them to make active decisions during the exercise. It was apparent, however, that some guidance was needed with first year students, so the exercise included eight general

questions about the site. These took the form of 'what are the proportions of coarse and fine pottery on the site and what does this mean?' and 'are there the remains of any structures on the site?' How many of these questions were answered, and in what detail was left to the students. A final question 'can you say anything else about the site?' was also included.

It was impressed upon the students that they should take their time to excavate the site, that they should understand what the budget meant in real terms, and that they should use the program's print facilities to make plans as the excavation continued. It was also explained that they should attempt interpretation as well as observation of their results, bearing in mind problems of confidence, certainty and bias due to their chosen excavation strategy. It was also impressed upon them that although they had a series of questions to answer, there were no absolute right or wrong answers, and that they were free to investigate anything they pleased within the restraints of the budget. After completing the exercise the students submitted a report.

The final reports submitted by students varied tremendously. Some chose to present a detailed discussion of both their strategy for excavation and the results obtained, while others took the path of least resistance and submitted answers to the eight questions and nothing else. This did not make marking the results easy. The final marks were intended to be representative not of the success in locating the archaeology, but of the level of thought given to the design of an excavation strategy, and on the level and quality of the interpretation. The aspects marked best were those which the students had been told to consider most closely, such as potential sources of bias in recovery and the nature of such bias; the problems of certainty in interpretation and the ways in which the chosen strategy for excavation may have effected the final interpretation.

The discussion prompted a variety of responses, not all positive. Of the negative responses, one of the most common was the that the exercise was not sufficiently guided. This was, I believe, the natural reaction of students educated mostly in a traditionally passive manner, suddenly faced with an exercise which required them to take the initiative. Many students, unsurprisingly, wanted to be told in more detail what to do and were unhappy with the uncertainty of deciding for themselves what to investigate. Other reactions from students were less concerned with the program philosophy (of which most seemed to approve) and more about the specific implementation. One of the most positive responses was a suggestion that the exercise should contain other types of information such as geophysical data, topography or results from auger sampling.

2.7 Implications of SyGraf

There are, I think, generally two models of the education process operating within archaeology and defining how the subject should be taught at University level. These can be characterised as teaching by passive learning, and teaching by active learning. In the former, the traditional form of education, the teacher is in control of the content and direction of the learning; the student is a passive receptacle

into which the wisdom of the teacher is poured. This form of teaching is characterised by lectures and traditional examinations or, at best, by demonstrations during which the student is required to passively receive information. This form of teaching is rooted firmly in the hierarchical nature of society and represents the way in which power is exerted within education. All control of the educational process in the passive learning model is in the hands of the lecturer, all decisions about the direction and content of learning are taken by the lecturer. This control of educational decisions not only imparts power to the teacher and denies power to the students but is also far easier for both the lecturer and the students. The lecturer does not have to respond to unforeseen situations, the student does not have to generate any original thought or provide any real motivation for learning. Lecturing has justifiably been characterised as the least efficient way of getting information from the notes of a lecturer to the notes of a student without passing through the minds of either. Although there is good lecturing and bad lecturing, all lectures are part of the passive learning paradigm.

There is, however, a growing awareness of an alternative to this passive process. SYGRAF attempts to contribute to a model of education which is characterised by an active student. In this educational model the student is given control of the decisions about what is learned, and how it is learned. This is not an easy task because traditional modes of education are seductive and, in their own terms, extremely successful — students who have sat doggedly through a series of lectures can retain the right information to pass examinations. If the mould of passive learning is broken, however, then the means of judging the success or failure must also change and it is my own belief that students who have been required to accept responsibility for decisions about their own learning process, and who have been allowed to actively participate in the process will make better researchers, teachers and citizens.

SYGRAF is nothing but a tool, and a prototype tool at that. This paper has described that tool and set out the motivation for building it. I hope (but by no means guarantee) that it reflects the views of all of those involved with the SYASS project. As a tool, however, SYGRAF can be used well or

it can be used badly. It can be used to perpetuate passive learning, should those who control access to it and create exercises to do with it choose so to do. I choose, however, to be optimistic and to claim that SYGRAF can contribute to an educational philosophy within which active students take decisions about their own education, and the rôle of the lecturer is altered from that of holder of power to facilitator and motivator. I do not claim that SYGRAF makes this easier, but it does perhaps suggest one way this can be approached. Chris Tilley has called for an 'archaeological glasnost'; is it then merely pure rhetoric to suggest that we first need an educational perestroika?

Bibliography

- BURROUGH, P. A. 1986. *Principles of Geographical Information Systems for land resources assessment*. Clarendon Press, Oxford.
- FASHAM, P. J. 1985. *The Prehistoric Settlement at Winnal Down, Winchester*. Monograph 2. Hampshire Field Club.
- HAWKINS, S. 1987. "An Examination of the Context of the Proposed Southampton-York Archaeological Simulation System", Syass Project Internal Document No. 6.
- O'FLAHERTY, B. 1988a. "The Southampton-York Archaeological Simulation System", in Rahtz 1988, pp. 491-497.
- O'FLAHERTY, B. 1988b. "The teaching of archaeological excavation—a new initiative", *Archaeological Computing Newsletter*, 15.
- RAHTZ, S. P. Q., (ed.) 1988a. *Computer and Quantitative Methods in Archaeology 1988*, International Series 446, Oxford. British Archaeological Reports.
- RAHTZ, S. P. Q. 1988b. "A resource based archaeological simulation", in Rahtz 1988, pp. 473-490.
- SHANKS, M. & C. TILLEY 1987. *Re-Constructing Archaeology*. Cambridge University Press.
- WANSLEEBEN, M. 1988. "Geographic Information Systems in archaeological research", in Rahtz 1988, pp. 433-51.
- WHEATLEY, D. W. 1989. "SyGraf—A Prototype Graphically Based Excavation Simulation for Computer Aided Learning", Master's thesis, Department of Archaeology, University of Southampton.