The Southampton York Archaeological Simulation System (SY_GRAF 3.54 documentation)

edited by Sebastian Rahtz

material by
Dave Wheatley
Brian Molyneaux
Sebastian Rahtz

March 1992

Contents

1

Introduction

Sebastian Rahtz*

This booklet documents the excavation simulation program, SYGRAF; this software was developed as part of the larger SYASS (Southampton York Archaeological Simulation System), funded between 1987 and 1989 by the Computers in Teaching Initiative of the UK Computer Board. The project was under the direction of Sebastian Rahtz & Steve Shennan (Southampton) and Julian Richards (York).

This documentation has been published by the Computers in Teaching Initiative Centre for History with Archaeology and Art History, University of Glasgow, 1 University Gardens, Glasgow G12 8QQ, UK (email `ctich@uk.ac.glasgow`) who will sell copies on request, and supply copies of the program. The documentation is also distributed as PostScript files as part of the distribution.

This version documents SYGRAF version 3.54. Changes from earlier versions of the documentation and program are marked with 'changebars' the margin (like this paragraph).

The Southampton York Archaeological Simulation System (**S**outhampton **Y**ork **A**rchaeological **S**imulation **S**ystem (SYASS)) was a joint project between the Universities of Southampton and York. It aimed 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 hoped to encourage awareness of computer technology amongst both students and staff.

A number of similar products already existed 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 were aimed at schools, and not at universities. Fugawiland, while strong in some respects, was 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 1988 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. Further details of approaches to the simulation problem were discussed in O'Flaherty, 1988a, O'Flaherty, 1988b.

The SYGRAF program itself developed from a short research project undertaken by David Wheatley as part of the M.Sc. in Scientific Archaeology (Archaeological Computing) at Southampton University in 1989. The program was intended as a re-write of the existing SYASS program, but concentrating effort on producing an intuitive and attractive interface (Wheatley, 1991).

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. It was enhanced again by Brian Molyneaux and Sebastian Rahtz in 1990 and 1991.

The SYASS project has reached the end of its funding, and the final product, David Wheatley's SYGRAF is now in a position to be distributed to UK archaeology departments for, hopefully, an extended period of use. This manual documents that program, and gives some background to the project. It contains

1. The installation guide for SYGRAF the program.
2. The users guide for SYGRAF, intended to be given to any student using the program.
3. Teacher's notes for SYGRAF; this provides suggestions for how the program can be used in the context of an archaeology course.
4. The technical manual for SYGRAF, needed by those intending to create new sites for excavation.

* Department of Electronics and Computer Science
University of Southampton

Acknowledgments

The SYASS project worked from 1987 to 1989 under a management committee consisting of Martin Carver, Tim Champion, Robin Torrence and Todd Whitelaw, together with the project leaders. We are very grateful to them for their advice and help.

Brendan O'Flaherty was the first research assistant, and he was responsible for all the initial setting-up and research. David Wheatley undertook to implement a complete system as an MSc dissertation, and subsequently as a research assistant, and must be be regarded as the great begetter of SYGRAF in every way. Brian Molyneaux worked again on the program in 1990, and provided more of the teaching insight. We thank these three for their belief in the concept, and their hard work.

Nigel Gardner, as the head of the Computers in Teaching Initiative in 1987, encouraged the project from the start; without him we would have achieved nothing.

The current release of SyGraf includes two programs from the Free Software Foundation, the Unix utilities `mv` and `tail`. You may copy these as you please. The program `delete.exe` was written by Eberhard Mattes for his release of the TEX package.

Bibliography

O'Flaherty, B. (1988a). The southampton-york archaeological simulation system. In Rahtz, 1988a, pages 491–497.

O'Flaherty, B. (1988b). The teaching of archaeological excavation—a new initiative. *Archaeological Computing Newsletter*, 15.

Rahtz, S. P. Q., editor (1988a). *Computer and Quantitative Methods in Archaeology 1988*, number 446 in International Series, Oxford. British Archaeological Reports.

Rahtz, S. P. Q. (1988b). A resource based archaeological simulation. In Rahtz, 1988a, pages 473–490.

Wheatley, D. (1991). Sygraf — resource based teaching with graphics. In Lockyear, K. and Rahtz, S. P. Q., editors, *Computer Applications and Quantitative Methods in Archaeology 1990*, number 565 in International Series, pages 9–14, Oxford. British Archaeological Reports.

2

Installation guide for SYGRAF

Sebastian Rahtz*

2.1 Requirements

2.1.1 Hardware

The use of SYGRAF requires an IBM PC, PC-AT or PS/2 or 100% compatible computer with *at least* 640k RAM, a Microsoft compatible mouse, EGA or VGA colour graphics card and monitor, and a hard disk.

Although SYGRAF will run on 8086 and 8088 machines with any hard disk, it is a slow program and greatly benefits from 80286 or 80386 based hardware, with hard disks faster than 40ms.

2.1.2 Software

The program requires MS-DOS or IBM PC-DOS, version 3.0 or higher (it may run on older versions, but this has not been tried). The config.sys file must also have the lines

```
files=20
```

and

```
buffers=10
```

(see the DOS manual).

SYGRAF is a CLIPPER program. It is highly memory-intensive, requiring most of the 640k DOS base memory. In order to avoid crashing the program because of lack of memory, it is recommended that the Clipper memory allocation be changed in the autoexec.bat file, or whatever batch file is used to drive SYGRAF. This may be done by adding the line `SET CLIPPER=V010;F70`. In addition, SYGRAF may not run with memory resident software like Sidekick and some TSR network managers. If problems occur, check that programs such as these have not been enabled.

SYGRAF will not run under Microsoft Windows in a window, but will work under Windows 3 in a full-screen DOS task; it will run under Desqview, but this has not been fully tested. To use SYGRAF under Windows 3, use the PIF file supplied (see your Windows manual), editing it if necessary to reflect your paths and file names.

2.2 Installing the program

SYGRAF is normally distributed on a single 3 1/2 inch DOS floppy disk; the contents of the disk are:

`install.bat`	program to install files on your hard disk
`sygraf.exe`	LHARC archive containing the main SYGRAF program, language files, and utilities
`site1.exe`	LHARC archive with database files for Greenacres site
`site3.exe`	LHARC archive with database files for UpHand Down site
`site3.exe`	LHARC archive with database files for Bush site

The 'install' program creates a directory on your hard disk called `SYGRAF` and installs all the files under that directory. It assumes that the floppy disk is drive A and the hard disk is drive C; if this is not true, you will have to use an editor and change the first few lines of the `install.bat` file. We will make this more sophisticated in a future release.

To use SYGRAF, the install procedure creates an example DOS batch file called `sygraf.bat` which sets the appropriate environment variables and starts the main program. This is what is documented in the user guide. You are welcome to create your own version of this, to create your own language version, for instance.

The installer of SYGRAF may need to do a small amount of setup, by changing the DOS batch file to run the program (`sygraf.bat`). Since version 3.54, the main program reads two DOS environment variables.

* Department of Electronics and Computer Science
University of Southampton

SYGRAF should be set to point to a complete drive/path under which the SYGRAF sites and utility files will live, while SYINFO should be set to simply the name of a file in the SYGRAF directory which contains user messages. The default installation utility puts all files under c:\sygraf, and sets SYGRAF to be c:\sygraf and SYINFO to be enginfo.txt. If the environment variables are not set, they revert to these values.

The effect of this change is to free the user from any reliance on having his/her default directory anywhere near SYGRAF; one can be established on a floppy drive, for instance. If the directory pointed at by SYGRAF is on the user's DOS path, the program can be used from anywhere. Files created in the Database section now therefore are created simply in the current directory and no longer insert 'A:' in front of the name as they did in older versions.

The other important side effect is that parameters to the 'sygraf' command are now interpreted differently from earlier versions. The first parameter is now the name of a valid site (eg site1), and the second parameter is now the name of a user in that site. If no parameters are given, SYGRAF looks around and builds a list of suitable sites; if no username is given, SYGRAF prompts for it, and offers a list of current userids from the chosen site from which to pick. Unfortunately, a bug in Clipper means that the scanning for site directories is not reliable — only the first 20 or so entries in the SYGRAF directory seem to be looked at. So if a site you know exists is not listed, give its name as the first parameter to SYGRAF.

2.3 Maintaining your SYGRAF

If you are using SYGRAF for student teaching, do not under-estimate the chances of files being corrupted, deleted, moved and changed, by accident or design. You should keep a fresh copy of SYGRAF on a floppy disk or on a 'safe' machine, and be prepared to reinstall files at intervals. The more enterprising students may change the information file, or use dBase to change their database files — there is little you can do to stop this.

2.4 Foreign language support

SYGRAF allows for use in any language which uses the ASCII character set. All the messages and prompts which the user sees are held in an external file which can be edited or replaced. This file is read by the program at the start of each run; which file is read is determined by the value of the DOS environment variable SYINFO. This should be set to the name of a file in the SYGRAF directory, eg SET SYINFO=ENGINFO.TXT. Currently only two language files are supplied, enginfo.txt (English) and portinfo.txt (Portuguese). Users are welcome to edit their copies of these files or translate them to other languages, but it is entirely their responsibility to ensure the file is in a valid format!! The format of file is a series of lines, each consisting of a number and some text, separated by a space. SYGRAF gives out messages by number — if you miss out a number or change the text, the program will look very odd! If you do make a new translation, please send a copy back to Southampton, so that it can be distributed with future releases.

2.5 Bugs

This is the fourth 'release' of the SYGRAF program. The first SYGRAF V 1.1 was written as part of David Wheatley's MSc course in 1989 and he issued a new version SYGRAF v2.1 in January, 1990, which fixed many of the bugs in the original. 3.43 (March 1991) incorporated a series of enhancements to David Wheatley's program, some of them major, involving both the graphics display and the database routines. Version 3.50, released November 1991, added the extra facility of export to Idrisi, and reworked the utilities. These enhancements were written by Brian Molyneaux and Sebastian Rahtz. None of the authors are aware of any bugs that will damage either the program or the rest of DOS, but we must rely on users to report bugs as they are found.

3

A User's Guide for SYGRAF

Dave Wheatley and Brian Molyneaux*

3.1 Introduction

The SYGRAF program is a laboratory-based archaeological site simulation, explicitly designed to be an interactive and open-ended learning environment for archaeology students. It consists of two main conceptual units, the resource (the site database) and the simulation program.

The resource is a DBASE relational database. By means of graphic routines, it creates a highly simplified representation of an archaeological site, consisting of contexts (archaeological objects which can be represented in plan by a polygon of some sort) and finds (which are represented as having only a position and not an area). These are arranged as a series of two-dimensional plans intended to represent differences in depth and, hence, the separation of archaeological components.

Any site may be represented in the system, as long as it can be documented according to the relational database structure. As each plan is two-dimensional, finds or contexts that occupy more than one excavation level present both practical and conceptual difficulties and should therefore be avoided if possible. The structure of the resource database, and the steps required to create a new resource database, is detailed in the *Teacher's Manual*.

The program allows the user to perform certain activities with the resource, primarily data extraction and analysis, that simulate the activities and decision-making in an actual archaeological situation. Most significantly, the simulation provides for 'open ended' exercises: within the constraints of a budget, the user is allowed to choose excavation areas anywhere on the site, select appropriate excavation methods (4 types of digging implement), and dig to any depth (to the default maximum of 9 levels). Using this approach, each excavation is unique.

The rules and limitations in this computer environment are, of course, those of the system rather than an archaeological site; the simulation is not meant as a replacement for real archaeological fieldwork. But these respective environments share the advantages of systematic work and provide penalties for bad practice.

The site simulation is intended, therefore, to be used to supplement the training of archaeology students before they encounter an actual site; it will make them more aware of the effect their decisions may have on the recovery of archaeological data. And it is not meant to be an end in its own right, but to be integrated into a curriculum as an interactive resource for the teaching of quantitative and descriptive archaeology.

3.2 Running SYGRAF

To start a SYGRAF session, the user should be in a suitable DOS directory where they wish to have any database work files created (see later), and then run the SYGRAF program, optionally supplying as parameters the excavation name and the username. Excavations are stored in subdirectories in the main SYGRAF area, not in the user's directory.

For example, if excavation of a site `site1` is intended, the procedure for entering an appropriate directory might be as follows:

```
C:\> cd work

C:\WORK>
```

To run the program, the user then types

```
C:\WORK> sygraf site1 name
```

followed by the return or enter key, where 'name' is the user's code name. This password allows each user to conduct an individual SYGRAF excavation and analysis programme, for the cumulative results of each session are stored in a subdirectory accessible by the user only.

Hence, users may work on the site according to their individual research strategies.

* Department of Archaeology
University of Southampton

If the system does not start, and instead the message `Microsoft compatible Mouse not detected---Install device driver` appears, then it means that either the mouse cannot be used with the system or, more likely, the mouse device driver software has not been installed. To rectify the latter the appropriate device driver must be installed (for IBM and Microsoft mice this means typing `mouse` at the DOS prompt). If the problem still occurs, the mouse is probably not Microsoft compatible, and it may not be possible to run the program. If the start is successful, the opening screen will then appear.

When the username has been entered, the program checks to see if it has been used before at the site. If the username is not recognised, either because the user is new, or because the username has been entered incorrectly, a second screen is presented, the User Options screen. The 'default' option (the option preselected by the program) is CREATE USER; a new user presses the return or enter key and is prompted to type a username in the space provided. The system then displays a message indicating that the new user subdirectory and database files are being created, and continues on to the main part of the program, the excavation area and the main menu. If the original username was entered incorrectly, the user can move the highlighted bar (using the arrow keys on the keyboard) to the appropriate label and re-enter the password—or quit the program altogether.

3.3 Returning to an Excavation in Progress

To re-enter the system from the DOS prompt, enter `sygraf site name`. The system will then display the opening screen, asking for the username. Enter the username exactly as entered when the system was first used and press return. The excavation will then be displayed with the excavation trenches, contexts and finds exposed from work in previous sessions.

If the system does not recognise a username that has been entered before, displaying the User Options screen (as above) instead, the username has almost certainly been mis-typed and should be re-entered.

3.4 The Excavation Screen

The SYGRAF screen (Fig. 3.1) consists of three areas:

The Site Plan The site area is the large green square that occupies most of the screen. At start-up it is overlaid with a white reference grid. All activity, other than database analysis, takes place directly on this screen: trenches are drawn and excavated, finds and contexts displayed, and information about these features retrieved.

The Menu This is the area to the left of the site plan which contains a vertical row of buttons identified by labels. Using the mouse, options can be selected by 'clicking' these (moving the mouse arrow to a button and pressing the left-hand button on the mouse) to perform the different activities associated with excavation. Some of the options perform actions, while others merely cause another 'sub-menu' to be displayed, with its own buttons.

The Message Area Beneath the menu is a blank area of screen. In this area the system will pass messages, either error messages (when the user has tried to do something which is not allowed by the program), information about the state of the excavation, or information about excavation costs and the budget. At startup, the current state of the user's budget is automatically displayed.

3.5 Using the Mouse

The mouse cursor is a small arrow. Moving the mouse on a flat surface beside the keyboard will reveal that the cursor mirrors the position of the mouse. The arrow cursor indicates that the system is ready for the user to point at something, either a menu button or an archaeological object.

When an action is selected, the cursor changes according to the state of the program. For example, while the system is accessing the disk drive or extracting information, the cursor may turn into an hour- glass. This means that the user should wait until the program has finished before doing anything else.

There are two buttons on the mouse (some may have three—disregard the middle button). Most of the work in the program is done by the left button. The right button is used only when the program is asking for an action to be confirmed (the **Confirm** button that appears in the menu). In this circumstance, the user may click the right hand mouse button instead of moving the cursor to the **Confirm** button in order to go ahead with the activity—saving a bit of time.

The mouse can be used for three different activities:

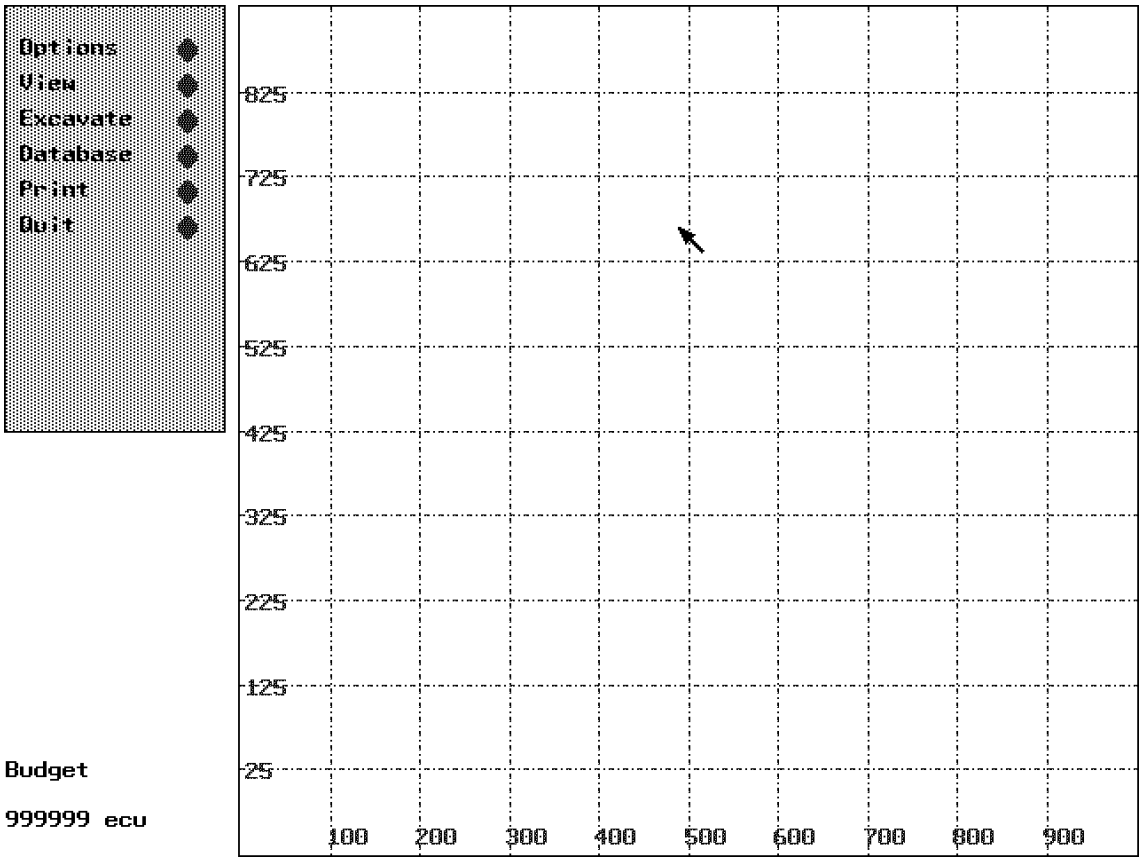

Figure 3.1: SYGRAF screen at start of new excavation

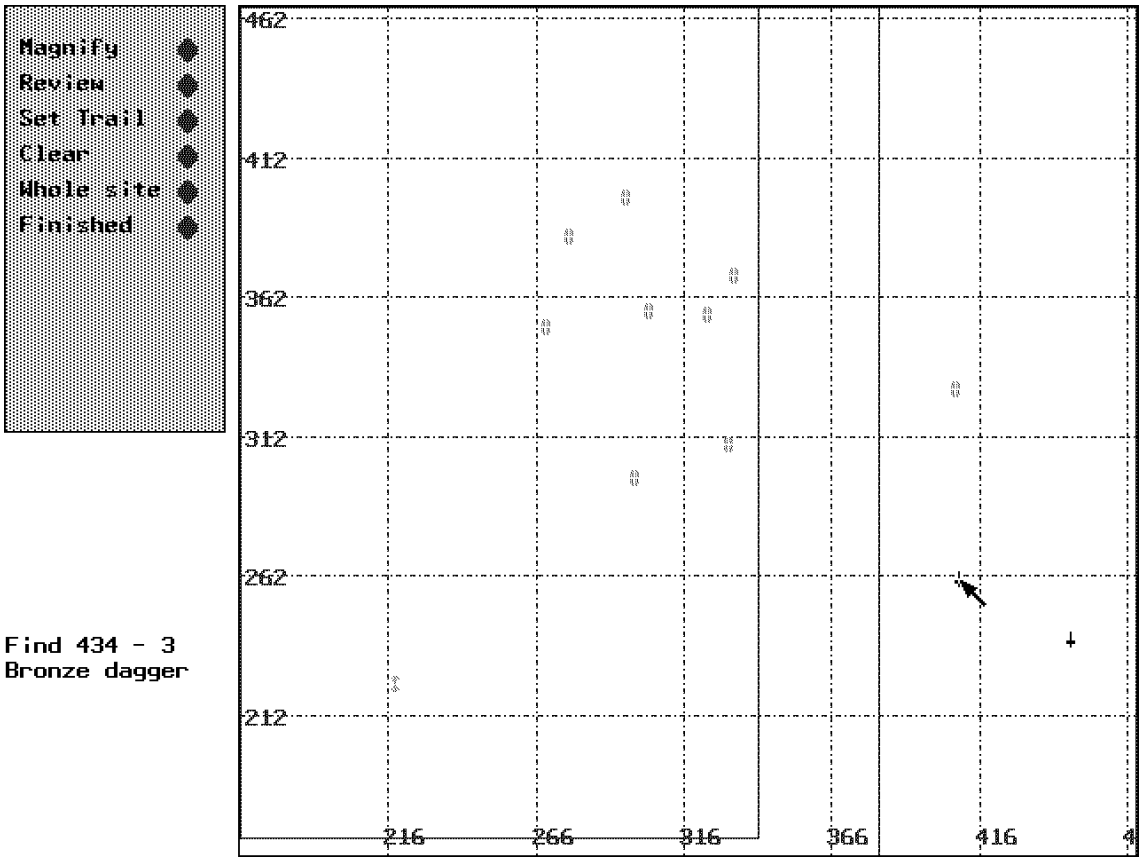

Figure 3.2: Clicking on a find provides details in bottom left corner of screen

Selecting from the menu By moving the cursor until the point of the arrow is on one of the buttons in the menu, and then pressing the left mouse button, the user can instruct the system to begin a process or change a program setting. Once 'clicked' in this way, the button will change colour to yellow while the program performs a task or prepares to display a further menu.

Defining an area If the cursor is a cross or a digging implement (see sections 3.7.3 and 3.7.2), the mouse performs a different function: defining an area of the site to be magnified or excavated. All areas are rectangular in shape, their sides parallel to the X and Y axes of the site. By positioning the mouse cursor at a point on the screen and then *pressing and not releasing* the left mouse button, a rectangle can be dragged out in any direction when the mouse is moved. Releasing the button ends the dragging operation and a prompt appears asking the user to confirm that the area is correct. Selecting **Confirm** from the menu (or pressing the *right* button on the mouse—see above) then tells the program to continue, while **Cancel** can be used when a mistake has been made.

Pointing at an object The contexts, finds and trenches displayed as graphics on the screen can also be interpreted by using the mouse. When the mouse arrow is clicked on the desired feature, descriptive information about the feature appears in the message area. This action may be done at any time, since it does not require the use of any of the menus.

Contexts can be identified if the **Tags** option in the **Options** menu is switched on. When this option is selected, a red square appears on each context as the spot to be clicked —the 'tag'.

Finds can be identified by clicking directly on the finds icon (the symbol representing a specific find) (Fig. 3.2).

The excavation area, trench number and number of levels in the trench can similarly be determined by clicking on the lower left hand corner of the trench. If there are finds or context tags in this area, it may be necessary to turn these off (using the **Options** menu) first. As the area sensitive to the mouse is also relatively small, it may be necessary to magnify the trench (see the **View** menu) in order to get this information.

If the cursor fails to detect information where it is clicked, the message *No Data Here* is displayed.

3.6 Digging the First Trench

The following is a description of the general procedure for digging a new trench. More detailed information on any of the specific actions or system features may be obtained from their specific entries elsewhere in this manual.

When the username has been accepted by the system and the necessary database routines have been finished, the main menu and site area are displayed. The excavation trenches are drawn directly on this area using the mouse.

The first step is to decide where on the site to dig and how large to make the trench. The user must imagine that as the site is a map or plan, the green site area is reduced in scale—a bird's eye view. Closeups of specific areas of the site may be obtained by using the **View** option on the main menu. This may be done prior to or after excavation; it should be noted, however, that it is not possible to distinguish small contexts and groups of finds if excavation is done when the screen is at the scale that displays the entire site.

As the program allows one to escape from routines by means of the **Confirm** procedure, the user may gain a better idea of the scale of the site by defining excavation areas (see below), which causes the system to display their potential cost, and then abandoning them.

As the excavation budget is limited, the user will then have to decide how to survey the site area in the most effective way.

When an appropriate area is selected, the user should click the **Excavate** option on the main menu and, when this appears, choose the excavation method by clicking the button beside the **Trowel** label, causing it to cycle (as the user clicks it) between the four digging implements.

To define a new trench, the user has two options: outlining the trench by hand with the mouse, or specifying precise coordinates.

To outline the trench by hand, the user must click the **New Trench** option, move the cursor across the screen to the excavation area and press the left-hand button of the mouse, holding it down. This establishes a point on the screen from which a rectangular box may be dragged in any direction (limited only by the site boundaries).

As the box is dragged out by the mouse, it appears as a slowly flashing white rectangle. When the box reaches the desired size and shape, the user releases the mouse button and the box outline turns red, indicating that the dimensions have been recorded. As long as the proposed trench is not too small—or larger than the budget will support—the confirmation menu appears and the message area displays the excavation level, the method, the potential cost, and the percentage of the current budget that it would use up (Fig. 3.3).

If the user decides to abandon the action, this may be done without penalty by clicking the **Cancel** option; the main menu will then be redisplayed. The red box will remain on the screen as a reference; you may ignore it when you drag out another box. If the excavation is to go ahead, the user may either press the right-hand button of the mouse (confirming the action automatically) or move the cursor to the menu and click the **Confirm** option with the left-hand button.

The user is cautioned to do this action with care: moving the cursor to the **Cancel** option and clicking with the right-hand button will cause the excavation to go ahead.

To dig by means of specified coordinates, the user must note the required minimum and maximum x and y coordinates by referring to the site grid and then click the **Set X/Y** option. Boxes for the two x and two y coordinates will appear on the screen. If any numbers appear already, simply type over them.

As the excavation progresses, the message area will relay the actions of the system: digging (Fig. 3.4), creating tags, sorting finds, and, when finished, the cost and new budget. At this point, the trench outline turns grey and the green interior is replaced by black. If any contexts or finds have been recovered, they will be displayed in colour in the trench, each different type of context or find in a different colour (Fig. 3.5).

The user may then continue the excavation to the next level, start a new trench, analyse recovered finds and contexts (see the **Database** menu), magnify or draw back from an area of the site (see the **View** menu), or alter the screen display (see the **Options** and **Database** menus).

The user is free to view any or all of the levels excavated, and to switch on or off the display of the excavation grid, contexts, finds and context tags (see sections 3.7.1 and 3.7.4).

If only one level of the site is shown, either because it is the only level, or all other levels have been hidden, the program will draw the contexts and finds using a different colour for each. For example, pits may be shown in light blue, ditches in green, graves in cyan, pottery in brown and bones in white. On the other hand, when more than one level is shown, all the finds and contexts of a particular level are given a single colour, so that the levels may be easily distinguished if they are superimposed.

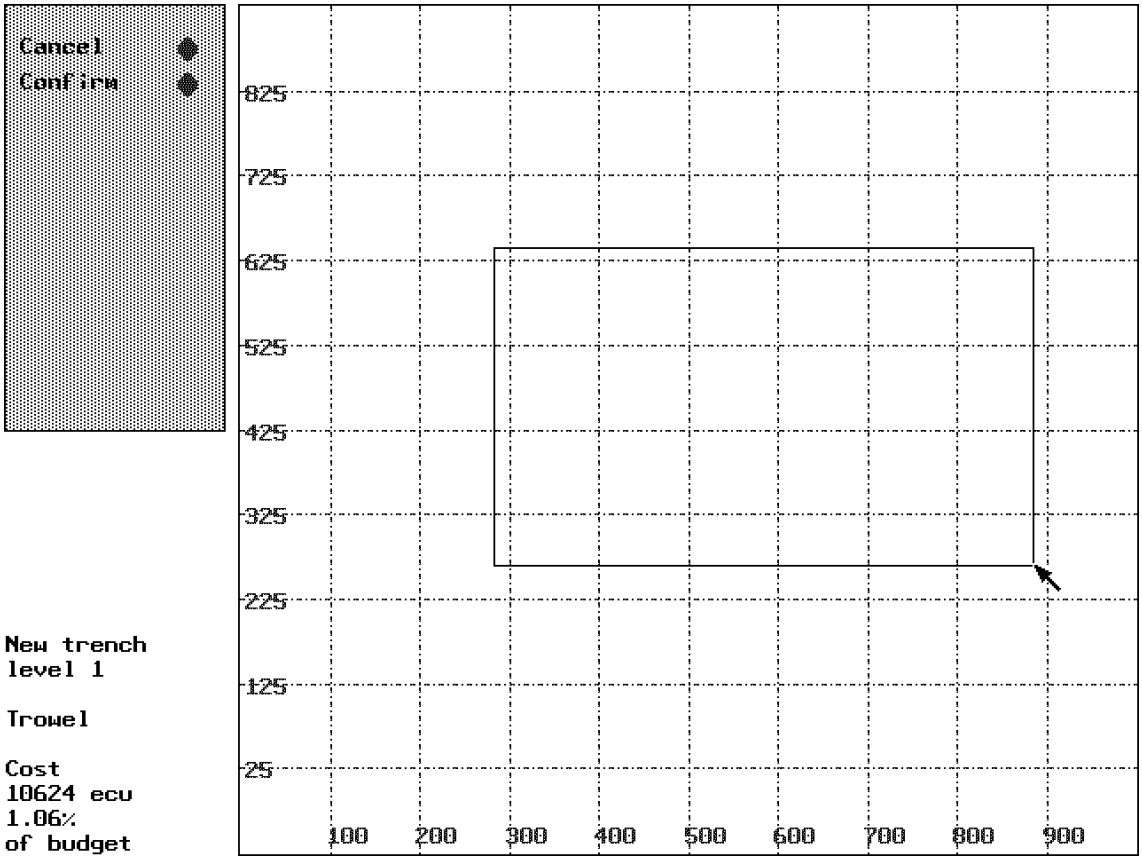

Figure 3.3: Trench defined, system waiting for confirmation

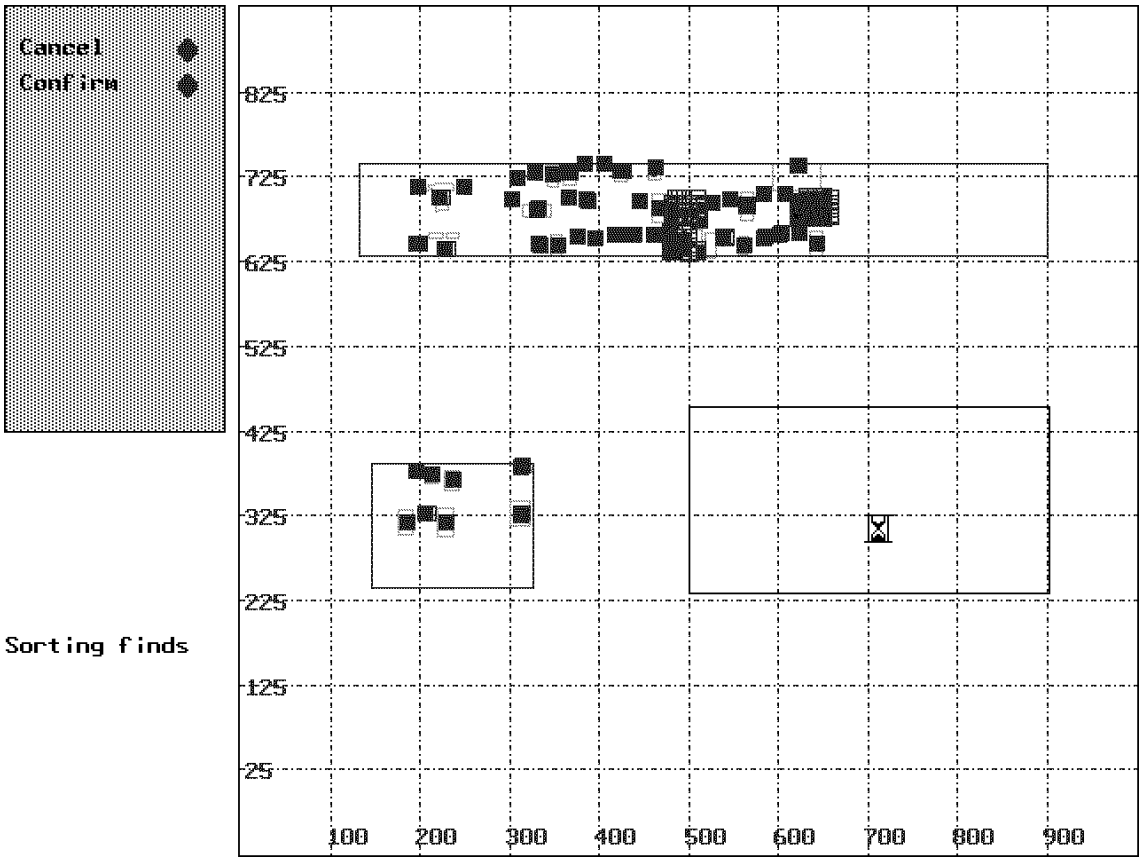

Figure 3.4: System in process of digging

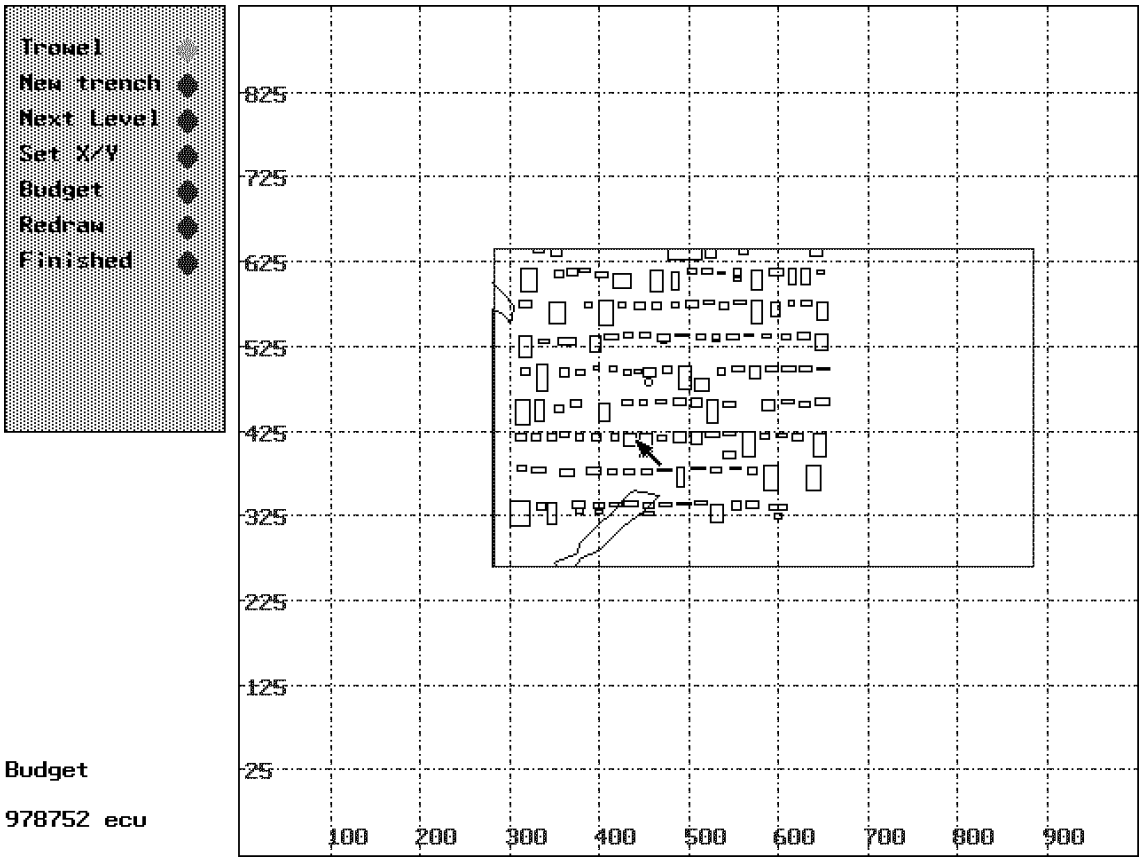

Figure 3.5: Completed trench

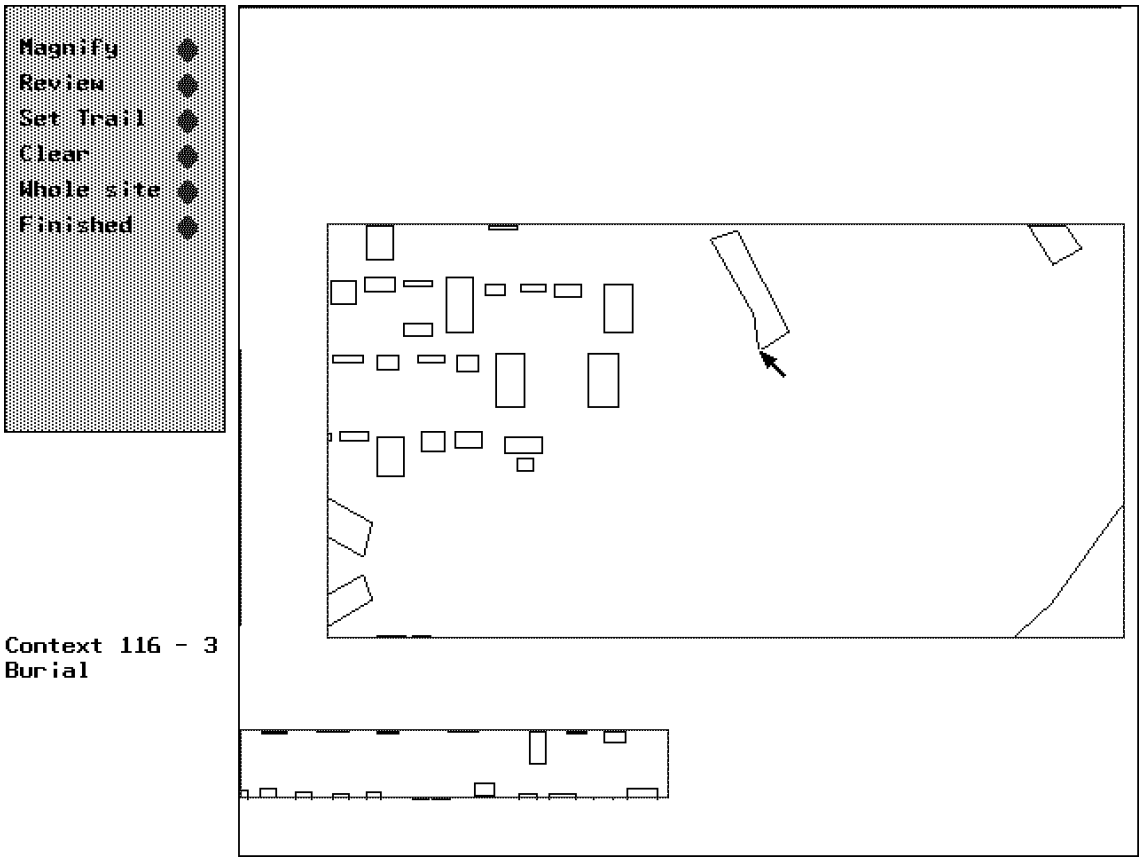

Figure 3.6: Excavation with grid turned off

3.7 Main menu options

3.7.1 Options

Selecting the options button causes another menu to appear on the screen. This menu contains six coloured buttons and two grey ones. The coloured buttons are *toggles* (switches). Clicking a button when it is off turns it on, and clicking it when it is on turns it off. The first four buttons are red when they are off, green when they are on and yellow when they are changing. The last two coloured buttons also change their label when they are switched. The options are as follows:

Grid Turns on or off the display of the dotted white lines and numbers which make up the reference grid. The grid which overlays the site is provided so that the user can identify the approximate grid reference of the objects. Switch it on to identify an object or to dig a trench in a particular place.

Finds Turns on or off the display of the icons that represent individual finds. As these are icons, representing the location of the find, their size is unaffected by changes in the scale of the plan— the icon remains the same size. This means that groups of finds may not be separated visually when the site is at its opening scale. The distribution of finds is more accurately pinpointed at greater magnifications.

Ctxts Turns on or off the display of contexts as coloured lines. Contexts are stored as polygons and drawn as a series of connected lines. As they are magnified, their polygonal nature becomes more obvious and they become more angular. This should not be taken as an exact represention of a context, but as an approximation of its shape.

Tags Determines whether a red tag is shown for each context, the tag being the spot on which the mouse may be clicked for information about the context. This toggle is normally switched off.

All Contexts *or* x Ctx Indicates whether or not a filtering condition defined in the database is inactive (a green button, labelled **All Contexts**) or active (a red button, labelled **x Ctx**).

If the toggle is switched to **All Contexts**, all the contexts in the excavated areas, regardless of excavation level or type, will be displayed at the same time.

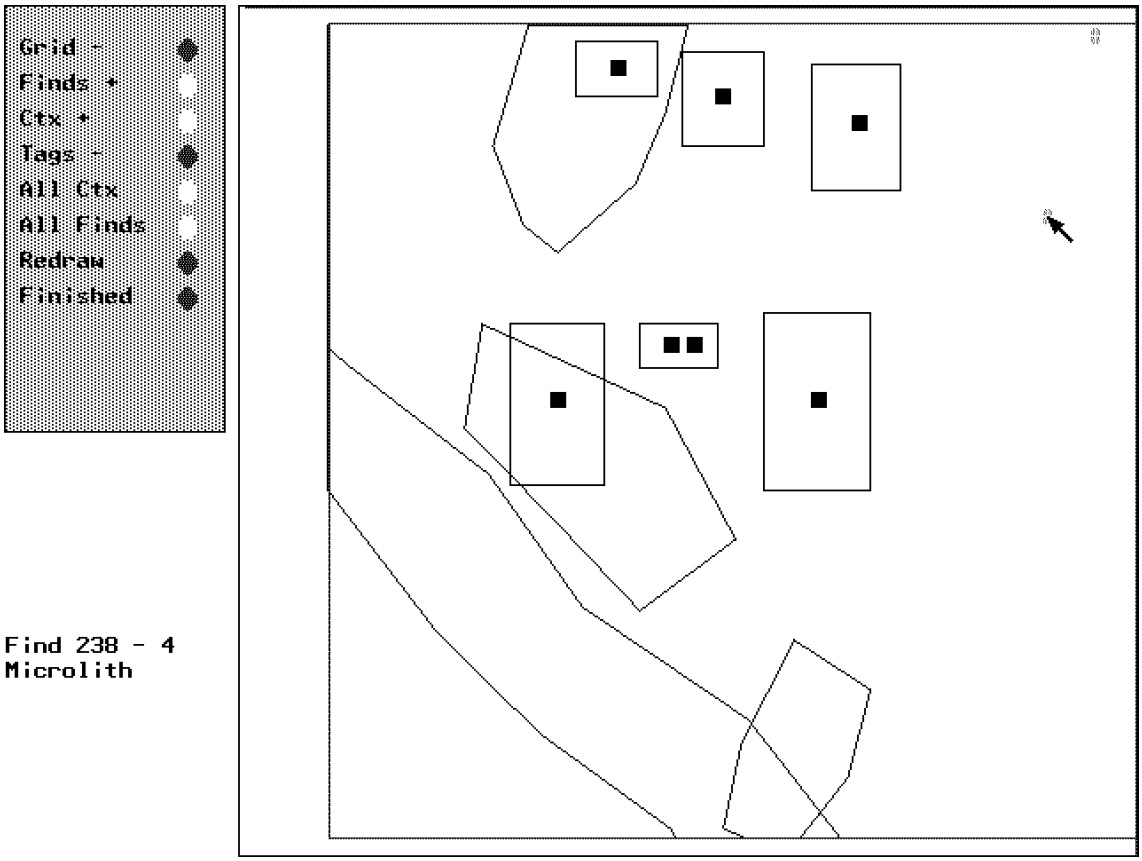

Figure 3.7: Magnified view of part of the excavation

If the toggle is switched to **x Ctx**, only the contexts of the specific excavation level defined in the Database will be displayed (see section 3.7.4 for more information).

All Finds or X Finds Indicates whether or not a filtering condition defined in the database is inactive (a green button, labelled **All Finds**) or active (a red button, labelled **x Fnd**).

If the toggle is switched to **All Finds**, all the finds in the excavated areas, regardless of excavation level or type, will be displayed at the same time.

If the toggle is switched to **x Finds**, only the finds in the specific excavation level defined in the database will be displayed (see section 3.7.4 for more information).

Redraw Enables the user to redraw the site after one of the toggles has been changed. After the redraw, the user returns to the same menu. This option is particularly useful in analysis if it is desired to see the effects of different filter conditions.

Finished Returns the user to the main menu. This option may be used in place of the 'Redraw' option if all work is completed, as it will automatically redraw the site if any toggles have been changed.

3.7.2 View

This option allows the user to change the area of the site displayed in the siteplan window. This magnification of a specific area reveals detail that may not be visible in the overall view. As with options, the **Finish** button returns the user to the main menu. The options are as follows:

Magnify Selecting this option causes the cursor to change into a cross, and to jump into the siteplan area. It cannot now be moved out of this area; it is waiting for a rectangle to be specified. Dragging a box with the cursor (see section 3.5) causes the program to redisplay only the area which is enclosed in the specified rectangle — creating a new, magnified display, a closeup (Fig. 3.7).

If the magnified area is a square, it should be replicated exactly because the siteplan area itself is square. On the other hand, if the magnified area is a rectangle, parts of the site additional to the area to be magnified will be included in the display to square it off.

If the area defined is too small (the minimum area is 10 units square), an error message will be displayed and the program will switch back to the main menu. The user may then attempt the closeup again.

Review This option is used with the **Set Trail** option below. If the button is clicked repeatedly, the program will cycle through the sequence of magnifications defined in the **Set Trail** option.

Set Trail This option enables the user to set up a series of magnified views of different excavation areas. By establishing such a trail, it is possible to move from one area to another without having to return constantly to the **Whole Site** display. The button is clicked and the user draws the required magnification area with the mouse. If more than one closeup is required, the **Continue** button is clicked. When the trail is set, the user clicks the **Finished** button. The trail may then be used by clicking the **Review** button.

Clear Selecting this option deletes all the previously recorded closeups defined by the **Set Trail** option. This allows the user to construct a new closeup sequence.

Whole Site This causes the program to show the whole of the site, as displayed at the opening of the program.

3.7.3 Excavate

Selecting the **Excavate** option from the main menu allows the user to excavate an area of the site—to dig a trench one excavation level deep (only one level can be excavated at a time). When the button is clicked, an excavate menu is displayed, containing one blue button and three grey ones. As with the other sub-menus, the **Redraw** option allows the user to refresh the site image without leaving the options menu and the **Finish** option returns to the main menu. The other buttons are as follows:

Trowel This blue button is similar to the toggle buttons on the **Options** menu, but has four states. At start-up it is set to **Trowel**, but clicking on it makes it cycle between **Trowel**, **Shovel**, **Pickaxe** and **JCB**. These digging implements are used to characterise the four excavation methods available.

These methods are conceived as the degrees of excavation precision possible with the different implements. This affects the recovery of finds, but not of contexts. If **Trowel** is selected, all the finds from the area excavated will be recovered; if **Shovel** is selected, fewer will be recovered; with a **Pickaxe**, fewer still; and if a **JCB** is used, no finds are recovered.

Against this, the user must consider the excavation cost, as there is a limited budget—and the more precise the recovery method, the more money will be spent. As a rough guide, it is approximately four times more expensive per cubic metre to dig with a **Trowel** than a **JCB**.

New Trench After the excavation method has been chosen, this option is used when a new trench is to be excavated. When the button is clicked, the cursor jumps into the siteplan area, where it will be restricted, and changes into an icon representing the digging implement to be used (trowel, shovel, pickaxe or JCB bucket). A box may then be dragged out to define the area to be dug (see section 3.5.

The size and location of the trench are subject to several specific conditions. The trench must be entirely within the site area (indicated by the green area of the plan); it must not overlap the borders of an existing trench—in other words it must be entirely inside or entirely outside an existing trench—either in a new area, or enclosing one or more previously excavated trenches; it must be greater than 10 units by 10 units in size; and there must be enough money in the budget to complete the excavation. If one of these 'illegal' conditions occurs, an appropriate error message is displayed and the routine is abandoned, returning the main menu to the screen.

When the desired area has been selected, the confirmation menu will appear, along with a message outlining the excavation plan: the area to be dug, the method, and the cost (in both currency units and as a percentage of the current budget).

If the planned excavation area encloses previously dug trenches (Fig. 3.8), the program accounts for these old trenches by adjusting the trench, finds, and context tables and the budget at the excavation level of the new trench. When the 'old' trenches are removed from the database, a 'deleting' message is displayed, indicating the trench number and level number as it is deleted. Any existing trenches that have been dug to the same level as the new trench enclosing them will disappear from the display; if an inside trench is deeper, however, it will continue to be an active trench, displayed in the new trench area, until the new trench is excavated to the same level.

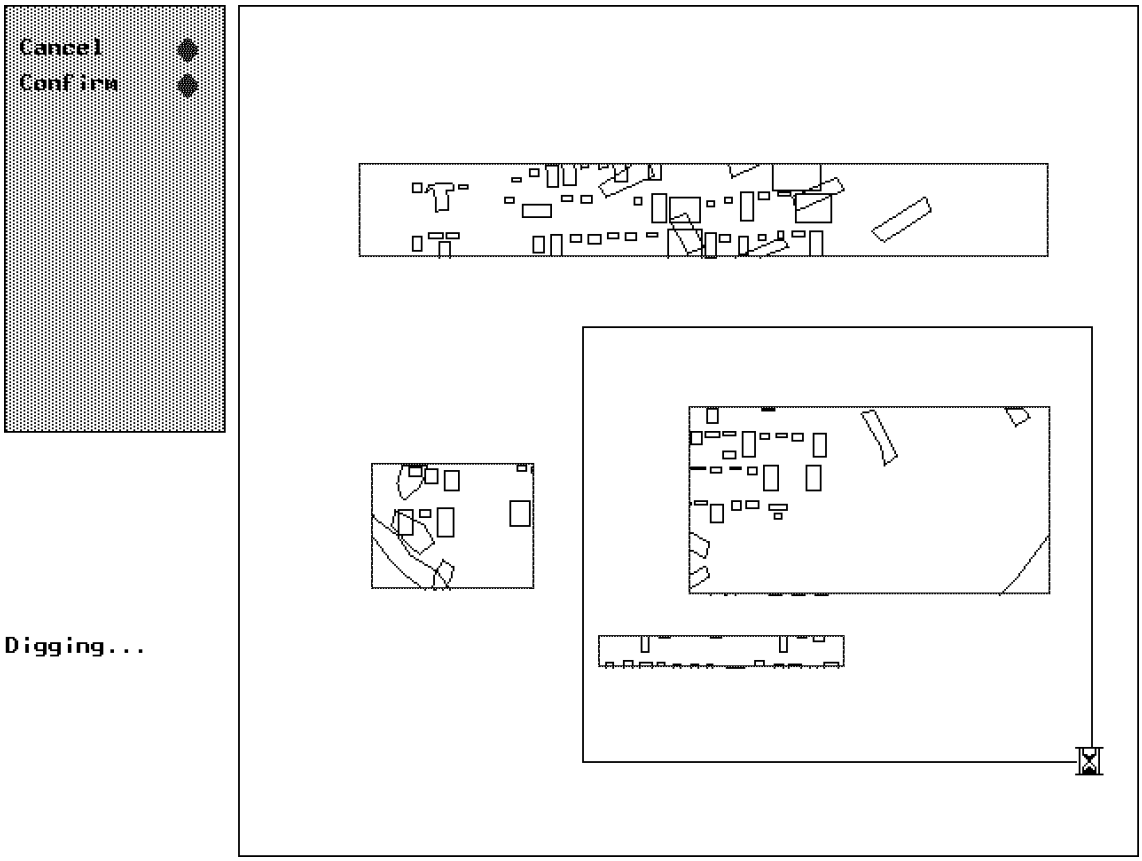

Figure 3.8: New trench which will remove traces of an earlier trench

Next Level This option allows the user to redig a previously excavated trench. After clicking the button, the cursor is displayed; the user must then move the cursor to the lower left hand corner of the trench to be excavated and click the button. The trench selected will become outlined in red and the message window will display the same questions and provide the same information as in the **New Trench** option.

Any trench may be redug, even if it is within other trenches, down to a maximum of 9 levels (the program default). If a tenth level is attempted, an error message is displayed and the user will have to find another option.

Set X/Y This option is selected when the user wants to excavate by means of specified coordinates. When the button is clicked, boxes for the two x and two y coordinates of a trench will appear on the screen. The user enters the desired coordinates and the excavation proceeds. If any numbers appear already, simply type over them.

Budget Selecting this button brings up the state of the user's budget in the message area. The budget is given in 'ecu'.

3.7.4 Database

Once excavation has been done, the user may access the database in order to select, sort and display or print lists of finds and contexts. When this option is selected, the site display disappears and is replaced by the database menu. This menu differs from the others in that the mouse is now disabled; the cursor keys must be used to move the highlighted bar to select options and the rest of the keyboard is used to enter information.

There are four options available, as follows:

Finds or Contexts Allows access to some or all of the finds or contexts data stored in the user's database. When one of these options is selected, a form is displayed which the user is required to fill in by entering data in the boxes. This allows the retrieval and display of information selected from the database by the user. The information may be displayed as lists of fields in the database or, with the **x Ctx** or **x Fnd** buttons enabled, as graphics on the excavation screen.

Back to Site This option finishes the current session with the database system and returns to the site display and the main menu.

Quit Sygraf This option finishes the current session, returns to the site display and then goes immediately to the DOS prompt.

If the **Finds** and **Contexts** data retrieval forms have been completed, the screen then offers a further four choices:

View Finds or View Contexts If this option is chosen, the data selected through the form is displayed on the screen. To view fields to the right of the window, move the highlighted box to the right with the arrow keys, to view records beyond the bottom of the screen move the box down with the arrow keys. To finish press the **Escape** key, which returns to the previous menu.

Write Ascii or Write DBF These options allow the user to write the data to floppy disk as either an Ascii file, or a *.dbf* (*dBase*) file. These files are written in the current directory. Enter a filename, which must be a valid *dos* filename (excluding the extension which the program adds

Display Site This option finishes the current session with the database system and returns to the site display and the main menu.

Finished This option brings back the preceeding Database menu, so that further Database options may be selected.

3.7.5 Using the Finds and Contexts Retrieval Forms

When either the **Finds** or **Contexts** option is selected, a data retrieval form is displayed on the screen.

Shifting between the boxes is done by moving the highlighted bar with the arrow keys on the keyboard. By typing data at the keyboard, the user can specify which aspect of the database is to be viewed.

The box labels are the names of the fields, with the only difference between the fields in the **Finds** and **Contexts** tables being the first box, which specifies the unique number of the find (Fname) and context (Cname).

Type is the field in which the code of the find or context is listed: 'p' for pottery, for example.

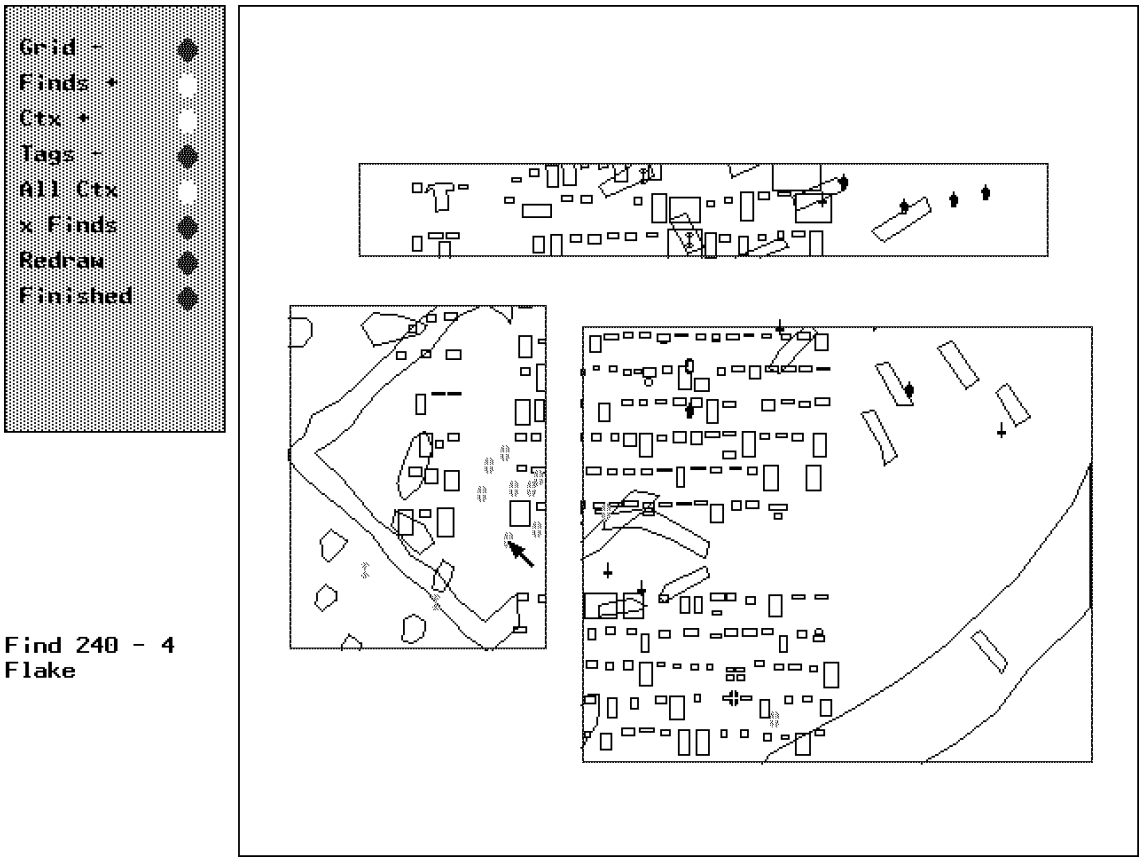

Figure 3.9: Excavation with all layers showing

Top Level and **Bottom Level** are the excavation level parameters which allow the user to select a specific level or levels for further analysis or display. The numbers in the boxes are the default levels, representing the minimum and maximum levels in the site. If these are left as they are, the user can display all finds or all contexts recovered from the excavations. Conversely, if both boxes are given the same number, only one level will be active.

Desc and **Desc2** are fields that contain the primary and secondary descriptions of the finds and contexts; these boxes are used if the user wishes to select a specific find or context type for display on the excavation screen or in the database mode. Typing "Post hole" in the **Desc** box, for example, would limit the graphic or text display to database records with that particular context. Similarly, one could limit the displays to "sheep" by entering this category in the **Desc2** box in the **Finds** table.

Sort on is an option that allows the user to sort, in alphabetical order, any of the fields in the database. To accomplish this, the user types the field name itself into the space. Sorting on **Desc2** in the **Finds** database, for example, will result in all the database records being arranged alphabetically according to this field.

The user will observe that there are additional fields in the **Finds** database. In the next version of the program, it will be possible to filter any of the fields in the same way as those above.

The filtering and sorting of the database is not limited to single entries; by typing the appropriate information in the boxes, the user may examine all the pottery from a specific level—level 2, for example—and have it sorted on the first descriptive field of the database. To do this, first select the **Finds** option, and then fill in the form as follows: Pass by the first box by pressing the down arrow, moving the highlighted bar and flashing cursor into the second box, labeled **Type**. Enter 'p' in this box (the code for pottery), then press the down arrow again. Enter '2' in the **Top Level** and the **Bottom Level** boxes, and then pass by the **Desc**, **Desc2** and the **Sort on** box. When this has been done, press the **Enter** key on the keyboard to carry out the retrieval. On returning to the graphical display, only those contexts or finds which meet the criteria are displayed — contrast Fig. 3.9, showing a site with all periods showing, and Fig. 3.10, which just shows layers 2 and 3.

Similarly, all the sheep bones (a category in the **Desc2** field) from level 3 (the **Top** and **Bottom Level** categories) in relation to post holes (the Desc field in the Contexts database) can be displayed as text — and, if the **x Finds** and **x Ctx** conditions in the **Options** menu have been clicked on, displayed in the excavation trenches that meet these conditions.

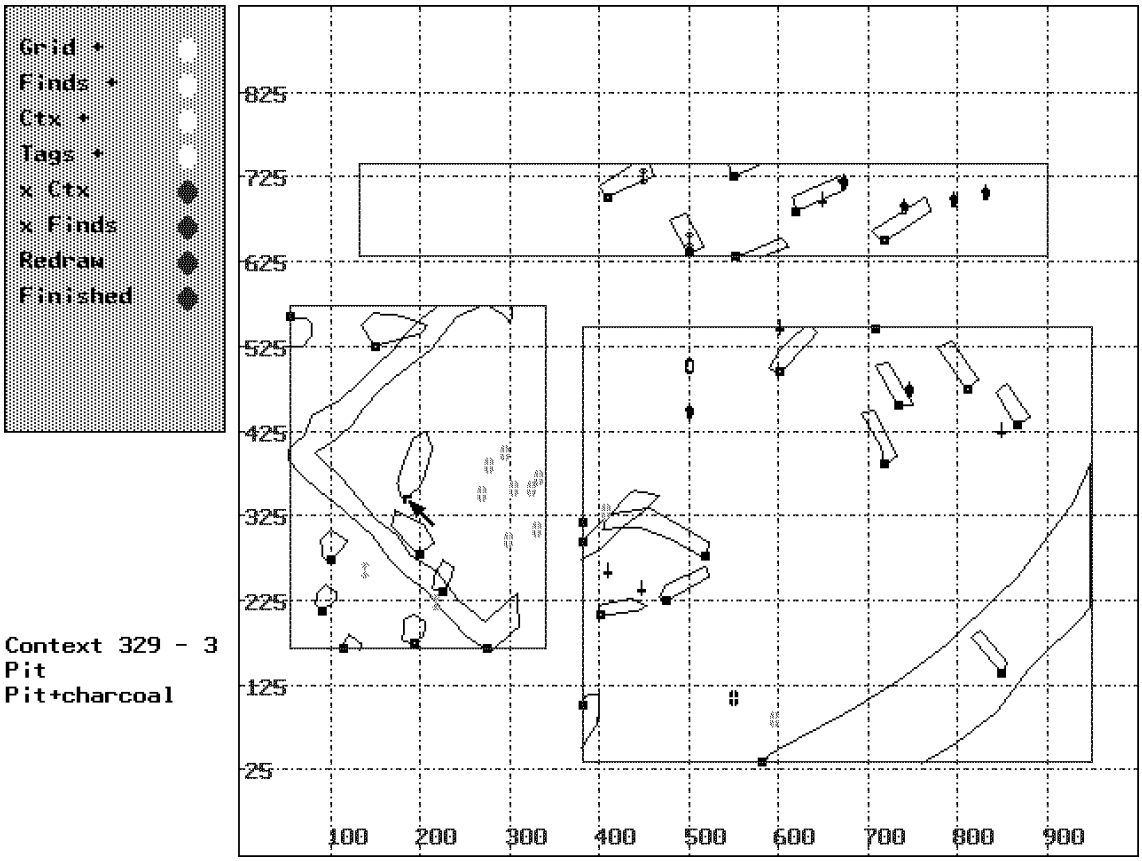

Figure 3.10: Excavation with only layers 2 and 3 showing

3.7.6 Addition — database export to Idrisi

Versions of SYGRAF after 3.50 allow the user to export datasets in a form suitable for use in Idrisi. This is a Geographic Information System package available from:

> Idrisi
> Graduate School of Geography
> Clark University
> 950 Main Street
> Worcester, MA 01610
> USA

at a price of $200 for individual educational users. It consists of a large number of DOS modules using a common file format; most of the work is done on raster images (such as site plans in our case), but the system also supports vector files, and has facilities for converting vector files into the internal raster format. SYGRAF writes Idrisi vector files, with the accompanying documentation file, and writes a DOS batch file with suggested Idrisi commands to convert them to raster images. The reader is naturally referred to the Idrisi manual for a full explanation of the whys and wherefores of all this, and the meaning of the following explanation.

The export facility is found in the Database section of SYGRAF, where 'Write Idrisi' is added to 'Write dBase' and 'Write ASCII' for any selected dataset. Finds and contexts must be exported separately.

Given a dataset, SYGRAF prompts for a filename (any valid DOS name), and performs the following actions:

1. An Idrisi environment file is looked for (`idrisi.env`) and read to establish paths and file suffixes.
2. An X/Y resolution is asked for, which will be the size of the resulting Idrisi images; the default is 590 $\times$ 450, which more or less fills a VGA screen.
3. A vector file is written (polygons for contexts, points for finds), with accompanying documentation file, in the Idrisi data directory. The identifiers are the IDs which have been assigned in SYGRAF.
4. A corresponding dBase file is written *in the current directory*, from which Idrisi 'value files' can be written for reclassifying the data. Thus contexts are initially given the value of their identifiers, but the user will want to later give them the values of eg context type or period.
5. Another Idrisi vector file called `TRENCH` is written, which consists simply of the rectangles which have been dug in the site.
6. In the *current directory* a DOS batch file is written which contains suitable Idrisi commands to perform the conversion to raster. To avoid the user having to create an initial empty raster, a dummy raster full of 0s is provided (`DUMMY.IMG` and `DUMMY.DOC` in the SYGRAF directory) and the batch file calls on `WINDOW` to extract a suitable blank image on which `POLYRAS` or `POINTRAS` can operate. A new title is given to the resulting image with a convoluted set of commands using programs called `mv` and `tail` which are provided in the SYGRAF directory (ie make sure that is on your path).

The more sophisticated user will simply look at the batch file and use it as a guide for future Idrisi operations.

The `SYTOOL` utility has an option to read the dBase file written from SYGRAF and write out an Idrisi values file. This is a partial replacement for the Idrisi `DBIDRIS` utility which seems unreliable.

3.7.7 Print

This option prints the currently displayed picture to a dot-matrix printer. By using the **View** option to display a particular area of the site, restricting the levels currently visible using the **Filters** options set in the Database (see section 3.7.4), and then printing the result, the user may build up a comprehensive collection of 'site plans' which may be extremely useful in later interpretation of the site.

To print the current view, ensure that the printer is switched on and is on line; then select the **Print** option. There will be a pause while the program re-draws the site on screen, and the printer will then reproduce the siteplan.

4

Suggested student exercises for SYGRAF

Dave Wheatley and Brian Molyneaux*

4.1 Exercise 1: by Dave Wheatley

4.1.1 A Description of the Exercise

This handout describes an exercise using the SYGRAF program. For this exercise you will use a site, partly invented though based on a real site, which is called UPHAND DOWN.

You will be given a budget which you must consider as you decide on a strategy for excavating the site. Keep a track of your budget, which is specified in Economic Community Units (ecu) of currency. Each ecu allows you to dig one square unit of one layer of the site with a JCB or one tenth of this area with a trowel (each unit of the site represents 20cm in real terms). You must balance the cheapness of digging at low levels of detail with the lack of information you will receive for doing this — if you dig with a JCB you will find no finds at all, though the outlines of the contexts will be revealed. If you dig with a trowel however, all the finds in the area are recovered in addition to the outlines of the contexts. In between these are two further levels characterised by a shovel, which recovers less finds than a trowel; and a pickaxe, which recovers fewer still.

When you have finished the excavation, in other words when you have too little money to continue to excavate, you must attempt to answer the questions at the end of this handout. It is essential that you read the whole of this handout before starting on the exercise. In particular you need to think about the best strategy for answering the questions *before* you start digging.

4.1.2 The Excavation

Keeping in mind the questions you have been asked to answer, the first part of the exercise is to excavate the site. Do not start this until you are happy using the mouse both to point at things and to drag boxes. If you are not happy with how the program is operated, practise on FakeSite (site1) until you feel more confident, this will remain available although users may disappear from time to time in order to keep as much space as possible free on the disk.

Remember to use your initials followed by the day of the month on which you were born as a username, there probably will not be any clashes this way. As you excavate you should bear in mind the following points:

Your Budget. You should decide in advance roughly what your budget represents in real terms, that is how much of how many layers you might be able to excavate, and be advised not to spend it all at once. There will be no more money.

Making Plans. The computer is connected to a printer (probably via a data switch) which allows you to make printouts of the plan on the screen, using this facility will allow you to take away a record of your excavation for consideration or as an illustration for the exercise.

Time. There is no limit on the amount of time you can spend on the exercise (subject to meeting the deadline given above), only on the area you can dig. If you want to, leave the program for a while, think about the problem and return later.

Size of Trenches. Obviously the larger the trench you dig, the more expensive it is. You should however bear in mind that the smaller the trench you dig, the more difficult it is to see what is 'going on'. There is a chance that the program will 'miss' context borders occasionally, and this chance is greatly increased with very small trenches.

4.1.3 Using the Site Database

In order to answer the questions below, you will need not only to examine the plans of the excavation and the picture on the screen, but also to use the site database system as described in the first handout. Not

* Department of Archaeology
University of Southampton

only will this give you more details about finds and contexts on the site, but it will also allow you to make quatitative summaries of the finds and context data. You will need to use the database both as you excavate, to obtain a better picture of what you have found; and when you have finished excavating, to formulate your interpretation.

If you are remotely unhappy with using the database system, read the example below and work it through with the program running in front of you. In the example, using the FakeSite (site1) database, it is assumed that you have already dug at least one trench and have recovered some of the the grave contexts in layer 1. The object is to explore the relative proportions of the different grave forms. It also assumes that you have read the section on the site database in the first handout.

4.1.4 A Worked Example

First select Database from the main menu, and wait until the database system has loaded. Next select Query Contexts to look at the contexts database. When the screen form appears type an 'g' in the Type box, and a 1 in each of the Layer boxes, then move the highlighted bar to the bottom box of the form (Sort on) and press return. The database will search for a short while and then present a screen of data (assuming that it has found anything matching the form).

Look at the column headed DESC, these are the grave forms. As you move around the screen, different forms become visible, and you gain an idea of which are the commonest and which are the rarest.

Next return to the main database menu by pressing escape, then selecting Finish from the menu bar. Do as before to retrieve all the graves in layer 1, but this time enter DESC in the Sort on box. When the data is shown this time, it is grouped into categories of grave form, and it is now possible to easily note down all of the different grave forms in the database, and to count the numbers of the ones with only a few entries. Write these things down.

On my excavation, the CROSS, PLG and HEAD forms are the commonest and there are too many to easily count. To obtain a count of each of these forms, use the database system again but further restrict the scope of the search to individual grave forms. For example, if you want to know how many graves there are with a HEAD form you could select Query Contexts, then fill in the form as above but enter 'HEAD' in the Desc box of the form. This will force the database to only extract those contexts of form HEAD, and give a count of the number of records extracted (in the top right hand corner of the screen). In my case it says 27, so I have 27 graves in layer 1 of form HEAD. Write down the result and repeat for all the grave forms you wish to count.

After all this you will have a table showing the relative frequencies of the grave forms in your excavation area, which you can use to begin to interpret the site.

4.1.5 Answering the Questions

When you have completed the excavation, and have as many plans as you wish to make, you must attempt to answer the series of questions which are listed at the end of this handout by using the information recovered by your 'excavation'. If you cannot answer one or more of the questions then you should explain why not, and attempt to justify the decisions which led you to a state from which you could not answer the questions. As you answer the questions bear in mind the following points:

Interpretation and 'Right' Answers. In the interpretation of your results you should bear in mind that there are rarely 'right' or 'wrong' answers in archaeological interpretation. This means that two different answers to the same question may be equally acceptable if they can each be justified well. One of the main problems of archaeological theory is how to decide between alternative interpretations of the same data, and much time and effort has been spent in recent years discussing how this should be approached and even if it should be done at all. Do not be afraid therefore, to attempt explanation of the site merely because you are afraid your interpretation will differ from mine; if you argue your case well you will receive good marks whether or not your interpretation coincides with mine.

Copying from Other People. Obviously if more than one person is using the program at any one time, it is tempting to look at their results and use them in your interpretation. There are, however, good reasons why you would be ill advised to copy other people's work or ideas. Firstly it will not help you in the long term because you will be making judgements based on data which I will be able to see you do not have. Secondly, as I have already said, there is no 'right' answer to many of the problems, so merely because someone else is employing different tactics to you, seemingly with more 'success', does not mean that they are obtaining the 'right' answer and you the 'wrong' one. You will lose vast numbers of marks for plagiarism.

How Confident are You? In other words how sure are you that the answer you give is a correct answer and, perhaps equally importantly, if you *were* wrong, how wrong do you think you could be? You may wish to discuss this in relation to the questions.

The Effects of your Decisions. What effect could your decisions have had on the results of the excavation, if you had made different choices could you have obtained different results? As you answer the questions you should consider possibilities such as bias in your excavation data due to the tactics you employed as you excavated.

4.1.6 Questions

1. How many occupation phases do you think there are at this site?
2. (a) Can you say what the extent of the occupation was in each of your phases?
   (b) Can you suggest a date for any or all of your phases? How?
3. What is the form of the structures at the site, how many are there in each of your phases and what do you think was their function?
4. Do you think there is any spatial differentiation within the site in any of the phases, for example areas of the site with different functions?
5. (a) What are the relative proportions of fine and coarse pottery at the site? Does this change through time?
   (b) Can you say anything about the spatial arrangement of the pottery?
6. Was the site enclosed at any time?
7. What species are represented in the animal bone collection and in what proportions? Can you suggest an interpretation? Does this change through time?
8. Is there evidence for any other activities not already mentioned?

4.2 Exercise 2: by Brian Molyneaux

4.2.1 Access to the Program

To enter the program, select the appropriate number on the computer menu or, if you are at the dos prompt, type: `sygraf site2 username` where 'username' is the name you have chosen for this exercise. It is recommended that you use your initials followed by the day of the month on which you were born as a username.

4.2.2 A Description of the exercise

You are faced with a grass-covered field, 180 metres $\times$ 200 metres, with no surface finds or features to indicate what may lie beneath.

Your goal is to discover as much as possible about previous cultural activity in this area by conducting an excavation. The success of your excavation will depend on the way you plan your excavation strategy and analyse your results.

You have been given a budget of 40,000 ecus (European Community Units) to carry out the excavation.

When you have completed the excavation and analysis, you must answer a number of questions (at the end of this handout).

4.2.3 The Problem

Excavation has a cost in time and money. These costs vary with the type of excavation tool used and the size of a given trench. As the budget is limited, it will be necessary for you to develop an excavation strategy that will give you value for money, so that enough information about the site may be obtained. When the money runs out, your excavation is finished.

Each excavation tool has a cost for its use. If you excavated the entire site area to a depth corresponding to Level 1, you would use up the following percentage of your initial budget:

JCB:	25%
pickaxe:	50%
shovel:	75%
trowel:	100%

If you decided to excavate the entire site with a trowel, for example, you would expose the entire Level 1, but you would not be able to dig deeper to determine if there were any earlier occupations.

Each tool also has limits on the amount of information it will recover. If you were able to excavate the entire site, each tool would expose all the contexts, but varying numbers of finds, as follows:

jcb:	no finds
pickaxe:	8% of total finds
shovel:	31% of total finds
trowel:	all finds

You must therefore choose your tools carefully, balancing the cost of exposing an area with the risk that information may be lost using a particular tool.

4.2.4 Digging the First Trench

The following is a description of the general procedure for digging a new trench. More detailed information on any of the specific actions or system features may be obtained from the SYGRAF Users' Manual.

When the username has been accepted by the system and the necessary database routines have been finished, the main menu and site area are displayed. The excavation trenches are drawn directly on this area using the mouse.

The first step is to decide where on the site to dig and how large to make the trench. The user must imagine that as the site is a map or plan, the green site area is reduced in scale—a bird's eye view. Closeups of specific areas of the site may be obtained by using the **View** option on the main menu. This may be done prior to or after excavation; it should be noted, however, that it is not possible to distinguish small contexts and groups of finds if excavation is done when the screen is at the scale that displays the entire site.

As the program allows one to escape from routines by means of the **Confirm** procedure, the user may gain a better idea of the scale of the site by defining excavation areas (see below), which causes the system to display their potential cost, and then abandoning them.

As the excavation budget is limited, the user will then have to decide how to survey the site area in the most effective way. It is advisable to develop a sampling strategy, in order to make your initial investigation as efficient as possible.

When the excavation approach has been worked out and the first excavation area is selected, the user should click the **Excavate** option on the main menu and, when this appears, choose the excavation method by clicking the button beside the **Trowel** label, causing it to cycle (as the user clicks it) between the four digging implements.

To define a new trench, the user has two options: outlining the trench by hand with the mouse, or specifying precise coordinates.

To outline the trench by hand, the user must click the **New Trench** option, move the cursor across the screen to the excavation area and press the left-hand button of the mouse, holding it down. This establishes a point on the screen from which a rectangular box may be dragged in any direction (limited only by the site boundaries).

As the box is dragged out by the mouse, it appears as a slowly flashing white rectangle. When the box reaches the desired size and shape, the user releases the mouse button and the box outline turns red, indicating that the dimensions have been recorded. As long as the proposed trench is not too small—or larger than the budget will support—the confirmation menu appears and the message area displays the excavation level, the method, the potential cost, and the percentage of the current budget that it would use up.

If the user decides to abandon the action, this may be done without penalty by clicking the **Cancel** option; the main menu will then be redisplayed. The red box will remain on the screen as a reference; you may ignore it when you drag out another box. If the excavation is to go ahead, the user may either press the right-hand button of the mouse (confirming the action automatically) or move the cursor to the menu and click the **Confirm** option with the left-hand button.

The user is cautioned to do this action with care: moving the cursor to the **Cancel** option and clicking with the right-hand button will cause the excavation to go ahead.

To dig by means of specified coordinates, the user must note the required minimum and maximum x and y coordinates by referring to the site grid and then click the **Set X/Y** option. Boxes for the two x and two y coordinates will appear on the screen. If any numbers appear already, simply type over them.

As the excavation progresses, the message area will relay the actions of the system: digging, creating tags, sorting finds, and, when finished, the cost and new budget. At this point, the trench outline turns grey and the green interior is replaced by black. If any contexts or finds have been recovered, they will be displayed in colour in the trench, each different type of context or find in a different colour.

The user may then continue the excavation to the next level, start a new trench, analyse recovered finds and contexts (see the **Database** menu), magnify or draw back from an area of the site (see the **View** menu), or alter the screen display (see the **Options** and **Database** menus).

The user is free to view any or all of the levels excavated, and to switch on or off the display of the excavation grid, contexts, finds and context tags (see **Options** and **database** sections).

If only one level of the site is shown, either because it is the only level, or all other levels have been hidden, the program will draw the contexts and finds using a different colour for each. For example, pits may be shown in light blue, ditches in green, graves in cyan, pottery in brown and bones in white. On the other hand, when more than one level is shown, all the finds and contexts of a particular level are given a single colour, so that the levels may be easily distinguished if they are superimposed.

4.2.5 Analysis and Interpretation

Making Plans The computer is connected to a printer (via a data switch) which allows you to make printouts of the plan on the screen. Using this facility will allow you to take away a record of your excavation for analysis or as an illustration for the exercise.

Using the Site Database In order to answer the questions below, you will not only need to examine the plans of the excavation and the picture on the screen, but also to use the site database system as described in the *Users' Guide*.

The database will give you more details about finds and contexts on the site, and will also allow you to make quantitative summaries of the finds and context data. You will need to use the database as you excavate, to obtain a better picture of what you have found, and when you have finished excavating, to formulate your interpretation.

4.2.6 The Exercise

The exercise is in two parts: recording the excavation and answering specific questions about your excavation results. Both parts are to be handed in.

Recording the Excavation An important part of this exercise is to chart your activities through time and space as you conduct the excavation. It is not simply the specific material results that count in this excavation, but the way that you have conducted it.

As you progress with your excavation, therefore, make a written record of every trench excavated, by writing its trench number and the time and date.

Underneath each entry, you may write a brief note as to the reasons for your decision to excavate in this place, using the particular tool you have chosen. These notes will help you later in the excavation process, as you follow your excavation plan, and they will provide the information necessary for the description of your research strategy and the interpretation of your results.

Answering the Questions When you have completed the excavation, and have as many plans as you wish to make, you must attempt to answer the series of questions which are listed at the end of this handout by using the information recovered by your specific excavation. If you are not able to answer a particular question, provide an explanation in terms of your excavation strategy.

Interpretation of Results In the preparation of your answers you should bear in mind that there are rarely 'right' or 'wrong' answers in archaeological interpretation. This means that two different answers to the same question may be equally acceptable if they are justified by the information in the respective excavations.

Sharing Information and Comparing Results with Other Excavators You may be tempted to look at the results of other excavations of this site and use them in your interpretation. Such information will not be helpful, because there is no single answer to be found, and no 'best' interpretation. SYGRAF and this exercise are more concerned with the *reasoning processes* involved in the approach to excavation and the analysis of the information recovered than the amount of material you have excavated. You will be successful if you carry out a careful excavation strategy, use your budget wisely, and answer the questions by reference to your own data.

The Questions

Append the log of your excavation and a printout of the contexts (structural remains and other features) and the finds recovered at the site to your answers.

1. What kinds of activities were conducted at the site?
2. Can you see any reason for the arrangement of the various features in an excavation area? Interpret their distribution and function.
3. How many occupation phases do you think there are?
4. Can you suggest a date for any or all of your phases? What is the evidence?
5. Determine the relative proportions of different artefacts at the site. Does this change through time? Why?
6. Can you see anything significant in the spatial arrangement of the artefacts? Discuss in relation to other site information.
7. What species are represented in faunal remains and in what proportions? Does the pattern change through time? Why?
8. Identify and interpret any patterns in faunal remains in relation to other site information.
9. Can you see any variation in the general pattern of occupation and activity over the site through time? Discuss.
10. Is there evidence for any other activities not already mentioned?

5

SY_GRAF technical manual

Dave Wheatley*

5.1 Introduction

This document describes how the SYGRAF site database is structured, how the program operates on the database, and therefore, how to construct an archaeological site database for the program to use. In order to construct a site database, a minimum level of knowledge is obviously required. We have not attempted, in this document, to write a fool-proof method with step-by-step instructions which anyone could follow. To create a database, then, you should have a working knowledge of the following:

1. Archaeology.
2. The SYGRAF program.
3. The MS-DOS operating system, commands, directory structure, naming conventions and utilities.
4. DBASE databases, indices and the DBASE language.
5. Databases generally, and particularly relational databases.

The program represents an archaeological site using a series of DBASE database files, these all have an extension of .dbf after their names. Thus when this manual refers to the 'contexts table', it means the file contexts.dbf.

In order to construct a site you do not, in theory, need a copy of DBASE though in practice you will find that life becomes extremely difficult if you do not have interactive access to your database files. If you have a copy of CLIPPER, you can use the RL and DOT utilities supplied with CLIPPER.

The best way to obtain archaeological data for creation of a site database would certainly be to record the site in the required format as it is excavated. This is currently an unreasonable expectation so the best alternative may be to use a digitising tablet to convert published material into DBASE files. There is no one standard way of storing digitiser output, so I have not attempted to describe in detail how you should operate this sort of hardware or how you should convert files to DBASE format. Instead I have opted to describe in detail the format of the site database, and have left it to the the reader to decide how best to put data in the required format.

5.2 The Site Model

The database is the implementation of a theoretical model of an archaeological site, this model represents a site in terms of objects and their properties (attributes). Each of the two primary kinds of object (contexts and finds) has attributes associated with it which identify the type of object it is, the layer in which it occurs and other things about it, depending on the type of object. In the case of finds, the database also identifies the level at which the user must excavate in order to recover it.

5.2.1 Contexts and Finds

The site is defined as consisting of only two kinds of primary object, contexts and finds. These may in turn be related to secondary objects such as points (which have no meaning in themselves but groups of which form the graphical description of a context) and layers. The term contexts is used within SYGRAF to represent both contexts and features, in other words anything found at a site which is *not* a find. This keeps the database as simple as possible so that there are not many tables for the program to search for information. This in turn means that the program retrieves data quicker.

Contexts are polygons, stored in the site database as a single record in the contexts table (see Table 5.2), and as a list of x,y points in the points table (see Table 5.4). This list of points represents the vertices of a closed polygon which approximates the shape of the context it represents. Each context of the real site must

* Department of Archaeology
University of Southampton

be reduced to this polygonal representation, and recorded with a set of attributes defined in the contexts table. Contexts have attributes which define their layer, type and descriptions. They also have four fields which describe a bounding box, which is the smallest rectangle with sides parallel to the x and y axes which totally encloses the context polygon. This is necessary to speed up the excavation and clipping routines. There are utilities to create closed polygons from open ones, and to build the bounding boxes from the list of vertices (see section 5.4).

When excavated, contexts are converted (by a clipping routine) into other polygons. If the same context occurs in different excavated areas, the single polygon stored in the site database is converted into several discrete polygons in the users database, each with a different unique identifier (the cname field). In this way, excavation of archaeological contexts such as pits, walls, trenches, ditches and so on can be simulated. The excavator has no sure way of knowing when, for example, he excavates the north end of a ditch in one trench and the south end in another, that the two resultant contexts are parts of the same feature. Converting the original site contexts into smaller discrete contexts, which have the same attributes but a different polygonal representation, leaves it up to the user to decide which contexts match up with each other.

Finds, on the other hand, are defined as only having a location; thus they are conceptually one dimensional as opposed to contexts which are two dimensional. They are far easier to deal with in a graphical model because they are always either inside or outside an area, never half and half.[1] Finds can therefore be defined by single entries in the table finds (see Table 5.5). When they are excavated, they are either moved to the user's finds table ufinds and given a new unique identifer (fname field) or not. They cannot be found more than once because the system does not allow excavation of any area more than once.

Finds have attributes which define which layer they belong to, their find type, the recovery level at which the user must dig in order to find it (coded numerically), and also a frame number attribute for use with the laser disk player (see section 5.5), if it is used.

5.2.2 Depth

The concept of depth of a site is simulated by the system by cheating. Instead of representing all of the contexts and layers as three dimensional objects, they are represented as two-dimensional ones (as plans) and assigned to a layer. The layers are stored in the layers table. Layers are numbered from 1 (at the top) to as many as nine (at present). Every time an area is excavated, only one layer is allowed to be dealt with. By controlling which areas an individual is allowed to excavate, (for example not allowing excavation of areas where the lower numbered level is not already excavated) the process of digging downwards is simulated.

5.2.3 Context_no and find_no as compared with cnames and fnames

These terms refer to the unique identifiers in the main and users database respectively. Because the program converts single objects in the main database into multiple objects in the users database, it invents unique identifiers for all objects (actually by taking the object type and suffixing this with the record number in the user's database). These new identifiers are not the same as the unique identifiers for objects in the main database. In the main database, the contexts are identified by their context_no, and finds by their find_no. In the users database, contexts are given a 'cname' and finds a 'fname' as unique identifiers. In each case there is an additional field in the users database which records the original context_no or find_no in the main database (this is so the program can refer to the main database tables for the descriptive attributes, and prevents excessive duplication of data).

5.2.4 Object types

Both contexts and finds have a type attribute, which is a one letter code defining what type of object they are. These codes are described in the sy_color table, and can be changed to suit any given site. The type identifier tells the program in what colour to draw the object (when only one layer is being viewed) and, in the case of finds, what icon to use.

5.3 The Database Format

5.3.1 Directory Hierarchy

Each site of the database is stored in a separate directory, beneath the directory which holds the SYGRAF program files.

[1] For the purposes of this model, finds can only be in one trench, they cannot occur between trenches. In real life Murphy's law says that all the most interesting objects always occur sticking out of the baulk

When the program is instructed to create a user, it creates a directory beneath the site directory, named after the user. It also adds the username to the list of names stored in the users table. After this it creates the empty database files ufinds, ucontext, and utrench in the users directory. It makes a copy of the defaults table called udefaults. This file is where the program records each user's excavation details.

5.3.2 Site Database

The site database consists of eight tables (.dbf files). All of these must be present with the right fields, and must contain data. The tables are described in Tables 5.2 to 5.12.

The contexts **table.** This must contain one entry for every context on the site. These contexts may be anything found at an archaeological site, except finds and layers, and must each have a unique context number (context_no); in all of my examples this comprises the context type followed by a number, though this is not essential so long as the context_no **is** unique. In addition, each context record must contain the layer (layer) in which it is to be found, the object type (type), the line drawing style (line), and at least two descriptive fields (desc and desc2). If desired, other descriptive fields (which may contain any data type) can be added to the table structure to better describe contexts.

The points **table.** The X and Y coordinates of every point on every context polygon must be stored in this table, together with the context_no of the context it belongs to and the order it occurs in the polygon.

An important characteristic of the points table is that the records must be grouped by context_no, and ordered by order for the program to run correctly. This breaks the relational database model, though the savings in time which allow the program to assume that the points are in order is well worth this minor transgression. The sytool utility will ensure that this is done correctly.

The finds **table.** This must contain a record for every find at the site. For each find, there must be an X and Y coordinate, a find type (type), the layer in which it occurs (layer) and the same two descriptive fields (desc and desc2) as for the contexts table. In addition, each find must contain a number from 1 to 4 in the rec_level field which represents the recovery level at or above which it will be recovered. Thus if a find has 4 in the rec_level field, it will only be recovered if the area is excavated with a trowel, if 3 then it will be recovered with a trowel or a shovel and so on. As with the contexts table, you may add fields to the finds table as you like.

The siteinfo **table.** This should have only one record, containing general information about the site, including the dimensions of the site itself (this determines the extent of the green area when the site appears on the screen, and where the users are allowed to dig). The table also contains the name of the site, which appears on printed plans.

The defaults **table.** This contains the default state of a users account when he or she logs on for the first time, it contains logical fields specifying the on/off state of the grid (grid), the find icons (finds), the context drawings (contexts) and the context tags (tags). In addition it includes the initial state of the budget field in the users defaults table (udefaults). This field, therefore, defines the budget allocated to the students for the excavation.

The sy_color **table.** This details all of the context and finds types, and specifies in which colors they should be drawn on the screen. You may wish to alter or re-write this table in order to accomodate mono vga screens, or to create new classes of find or context. Each find or context type must be a letter, and must be different from all the other letters (though you can have the same letter representing a context and a find). Each record of this table, then, must contain the character code, the objecttype (0 for finds or 1 for contexts), the color in which to show the find or context (given as the number of the color in the vga or ega 16 color palette) and (if it is a find) a number representing the icon to draw for the find. In addition there is a field for notes, which performs no function except as a reminder of what the code is meant to be. The icon codes and vga default palette numbers are given in figure 5.1.

The layers **table.** This describes each layer of the site, at present it is more elaborate than the program requires. It is sufficient to include one entry in this table for every layer, giving it a number (layernum) and the color in which it should be drawn (layercolor). There are also fields for a description of the layer (desc) and its soil type (soil_type).

The users **table.** This should be empty when a new site is created, so that the program can add usernames to it as new users log on for the first time. This table has a single field (username) which the program uses to remember who is a registered user.

Previous releases of SYGRAF insisted that there be a number of empty tables existing in the site directory for the program to run ('pattern' files which the program copied to the user's database when a new user was created). These are now created dynamically by the program. When a new excavation is started, the following tables are created:

The ucontext **table.** The ucontext table is the location of the user's context records, and the table which correlates cname with context_no, and records which layer and trench the context belongs to. In this table none of the descriptive fields in the main contexts database are repeated. It contains the cname, type, layer and line fields from the contexts table, and additional visible, trench, context_no, tagx and tagy used in the excavation process.

The ufinds **table.** This contains a record for every recovered find, with an fname, layer, type, the trench in which it occurs, and also the find_no of the find in the main database (so that the database and labelling routines can obtain the descriptions from the main finds table, preventing unnecessary duplication).

The utrench **table.** This is where a record of the dimensions of the trenches the user has excavated is kept, together with the layer in which the trench is dug. Each trench also has a record of which trench it is inside (recording 0 if it is a 'fresh' trench in layer 1). This is necessary for the logic allowing excavation within trenches.

In addition to the DBASE files, a special binary file called upoints.dat is created to hold the point information for each context. Each entry in the ucontext table contains a number which is a pointer into this file. This can only be accessed from the SYGRAF program which retrieves a whole array of points as fast as possible.

5.3.3 Index Files

The program also requires three CLIPPER index files to be present in the same directory as the database files. CLIPPER index files are **not** the same as DBASE index files, CLIPPER indices being differentiated from DBASE ones by using the .ntx suffix instead of DBASE's .ndx suffix. You cannot, therefore, create the indices for the site database using DBASE III+ or DBASE IV, but must either use the supplied sytool utility (see section 5.4) or use CLIPPER itself, (either the RL or dot utilities supplied with the compiler) to create the indices. I recommend that you use the sytool utility to ensure complete compatibility.

The three index files that the program requires are as follows:

ccontext.ntx Index to contexts table on the context_no field.

pcontext.ntx Index to the points table on the context_no field.

ffinds.ntx Index to the finds table on the find_no field.

5.4 Utilities

A utility program is supplied with the main SYGRAF program to make creation of site databases easier. It is not necessary to use it to obtain a database which conforms to the specification above, but using it will minimise the risk of errors, and should therefore speed up the final stages of preparing the site database. This program includes all of the utilities written to aid maintenance or development of SYGRAF sites, combined together in a single menu-driven program called SYTOOL which is installed in the same place as SYGRAF. The facilities available are:

1. Set all records of a user back to scratch, leaving a 'clean' excavation
2. Delete a user from a site completely
3. Delete *all* users from a site
4. Rebuild the main index files which SYGRAF expects to find when it starts (in the case of suspected corruption)
5. Read a dBase file produced in the Database section of SYGRAF and produce an Idrisi 'values file' (see Idrisi manual; this facility is similar to DBIDRIS, but converts values to integer, writing a separate file containing expansions in the form of 'legends' which can be added to a documentation file).

Number.	Icon drawn on screen.
1	Circle
2	Triangle
3	Dagger
4	Bone
5	Pot
6	Flint
7	Nail
8	Coin
9	Square

Number.	Color in default palette.
1	Black
2	Dark blue
3	Green
4	Cyan (pink)
5	Red
6	Magenta (light blue)
7	Brown
8	White
9	Grey
10	Bright blue
11	Bright green
12	Bright cyan
13	Bright red
14	Bright magenta
15	Bright white

Table 5.1: Icon and Color codes for use in the sy_colors table

FIELD	TYPE	PURPOSE
CONTEXT_NO	*char 5*	Unique identifier
TYPE	*char 1*	Type identifier
LAYER	*num 2*	Layer number
BBMINX	*num 8*	Min. x of bounding-box.
BBMINY	*num 8*	Min. y of bounding-box.
BBMAXX	*num 8*	Max. x of bounding-box.
BBMAXY	*num 8*	Max. y of bounding-box.
DESC	*char 15*	Compulsory description field.
DESC2	*char 15*	Compulsory description field.
LINE	*num 1*	Line style for drawing outline.
Other	*any*	Optional additional fields.

Table 5.2: Logical structure of the contexts table

FIELD	TYPE	PURPOSE
CNAME	*char 5*	Unique identifier.
TYPE	*char 1*	Type identifier.
LAYER	*num 2*	Layer number.
TRENCH	*num 4*	Trench number.
CONTEXT_NO	*char 5*	Context number.
TAGX	*num 4*	X coord of tag.
TAGY	*num 4*	Y coord of tag.
LINE	*num 1*	Line style for drawing outline.
REC_LEV	*num 1*	Recover level (not used).
VISIBLE	*logical*	internal flag for controlling display.
NPOINTS	*num 3*	Number of point records.
POINTPOS	*num 6*	Pointer into upoints.dat file.

Table 5.3: Logical structure of the ucontext) table

FIELD	TYPE	PURPOSE
CONTEXT_NO	*char 5*	Context identifier.
X	*num 8*	X-coordinate of vertex.
Y	*num 8*	Y-coordinate of vertex.
ORDER	*num 3*	Order of vertex in list.

Table 5.4: Logical structure of the points table

FIELD	TYPE	PURPOSE
FIND_NO	*char 6*	Unique identifier.
LAYER	*num 2*	Layer number.
X	*num 8*	X-coordinate.
Y	*num 8*	Y-coordinate.
TYPE	*char 1*	Find type.
REC_LEVEL	*num 1*	Min. level at which it is found.
FRAME_NO	*num 5*	Frame number on laser disk (if any).
DESC	*char 15*	Comulsory description field.
DESC2	*char 15*	Comulsory description field.
Other	*any*	Optional additional fields.

Table 5.5: Logical structure of the finds table

FIELD	TYPE	PURPOSE
FNAME	*char 6*	Unique identifier
TYPE	*char 1*	Find type
LAYER	*num 2*	Phase
X	*num 8*	X-coordinate
Y	*num 8*	Y-coordinate
FIND_NO	*char 6*	Unique identifier.
TRENCH	*num 4*	Trench number.
VISIBLE	*logical*	internal flag for controlling display.

Table 5.6: Logical structure of the ufinds) table

FIELD	TYPE	PURPOSE
SITEMINX	*num 5*	Minimum x-co-ordinate of site.
SITEMINY	*num 5*	Minimum y-co-ordinate of site.
SITEMAXX	*num 5*	Maximum x-co-ordinate of site.
SITEMAXY	*num 5*	Maximum y-co-ordinate of site.
NAME	*char 20*	Name of site.

Table 5.7: Logical structure of the siteinfo table

FIELD	TYPE	PURPOSE
USERID	*char 8*	Unique Identifier of user
others		Undocumented and unused as yet

Table 5.8: Logical structure of the users table

FIELD	TYPE	PURPOSE
FINDS	*logical*	Whether to show finds at startup.
CONTEXTS	*logical*	Whether to show contexts at startup.
GRID	*logical*	Whether to show grid at startup.
TAGS	*logical*	Whether to show tags at startup.
BUDGET	*num 6*	Amount of money each user is initially awarded.
GRASSCOLOR	*num 2*	Colour of background (not implemented)
MENUCOLOR	*num 2*	Colour of menu (not implemented)

Table 5.9: Logical structure of the defaults table

FIELD	TYPE	PURPOSE
CODE	*char 1*	Letter code for object.
COLOR	*num 2*	Colour to draw object.
ICON	*num 2*	Icon type to use for finds.
OBJECTTYPE	*num 1*	0 for find, 1 for context.
NOTES	*char 20*	Notes.

Table 5.10: Logical structure of the sy_color table

FIELD	TYPE	PURPOSE
AREA_NO	*num 4*	Unique number of trench.
MINX	*num 8*	Min. X coord. of trench.
MAXX	*num 8*	Min. Y coord. of trench.
MINY	*num 8*	Max. X coord. of trench.
MAXY	*num 8*	Max. Y coord. of trench.
LAYER	*num 2*	Layer trench is cut into.
INSIDE	*num 2*	Inside which other trench, 0=none.
EXCOST	*num 9*	How much did it cost to dig.
TRENCHNUM	*num 4*	Internal counter.
DATE	*date 8*	Date of excavation (unused).
TIME	*char 8*	Date of excavation (unused).

Table 5.11: Logical structure of the utrench) table

FIELD	TYPE	PURPOSE
LAYERNUM	*num 2*	Layer number.
SOIL_TYPE	*char 15*	Soil type of layer.
DESC	*char 40*	Description.
LAYERCOLOR	*num 2*	Colour to display layer.

Table 5.12: Logical structure of the Layers table

It is suggested that this program is not generally advertised to student users.

In the case of the first 4 utilities, the name of the site can be given as a parameter to SYTOOL. The Idrisi utility will expect to find an idrisi.env file in the current directory. The options which remove user IDs need access to a program called delete.exe which is provided in the SYGRAF directory (put it on your path).

There is an extra 'hidden' utility which gathers together the various routines which are normally used to clean up a freshly-created SYGRAF site (indexing, checking for closed polygons, building bounding boxes etc). This is *only* available if the first parameter to SYTOOL is NEW and the second parameter is the name of a valid site.

The source of SYTOOL is given in the documentation directory of SyGraf to allow developers of sites to alter the routines *ad lib*. Compilation requires a copy of Clipper 5.1.

When a new site has been created, the author should use the NEW parameter to SYTOOL and the site name, and the following tasks will be performed.

1. We convert 'open' polygons in the points table into 'closed' polygons. Depending on the source of the context data, it may well be that the lists of vertices in the points table describe polygons whose first and last points are different (some software will assume a closed shape and so will join the first and last points together). The SYGRAF program requires that the first and last points of each polygon be identical.

   When the program is run, it searches for a table called points and one called contexts. These must conform to the site database specification. When the program finds these tables, it will compare the first and last points of each context and if they are different, add another copy of the first point to the end of the list. The program also checks whether any contexts exist in the contexts table but not in the points table. If this happens, the program prints an error message.

2. We create the bounding box fields in the contexts table. This works by identifying the same tables as for buildctx, and then for each entry in the contexts table, it checks the list of vertices in the points table, entering the dimensions of the bounding box in the four fields of the contexts table.

3. As described above, the program relies on three index files to the site database, pcontexts.ntx, ccontexts.ntx and ffinds.ntx. These are built automatically once the database is complete.

5.5 Laserdisk Player Use

5.5.1 Description and Warning

The program currently has limited built in routines for accessing a laserdisk disk drive connected to the serial communications port of the computer (comm 1). These routines were written for the Philips VP410 Laservision Disc Drive, and have not been tested on any other equipment. It should also be stated this is experimental, and there is therefore **no error-checking** incorporated into these routines. **This is a serious warning**, we accept no responsibility for anything that happens to you as a result of using the laserdisk interface, and have only documented it out of a sense of thoroughness.

The interface allows for finds to be illustrated with one still-frame picture stored on the laserdisk. This picture is displayed on a monitor connected to the videodisk player when the user clicks on a find icon. To enable this to work, the frame_no field in the finds database must be completed, so that each find includes a record of the number of the frame on the videodisk.

5.5.2 How it works

The videodisk interface is disabled by default, and must be switched on by using a command line option from the DOS prompt as follows:

```
c:\sygraf -v
```

Where the -v option switches on the videodisk interface, and causes the program to assume that a videodisk player which understands what it is saying is connected to the comm 1 port. **Do not use the videodisk interface if anything except a videodisk player is connected to the comm 1 port of the computer**, if you do weird things may happen to you (they did to me when I tried using a serial mouse connected to comm 1).

When a user clicks on a find icon, the program sends a character string to the videodisk. This character string is the correct F-code sequence for the Philips disk drive (and I guess any that have the same interface) to go to a single frame and display it.

5.6 Glossary

Bounding-box The bounding box of a polygon is defined as the smallest rectangle with sides parallel to the X and Y axes, which totally encloses it.

Clipping The process of converting one polygon into a smaller polygon by altering its shape so that only areas defined to the clipping algorithm are recorded. In SYGRAF this involves comparing polygons with trench rectangles, and altering the vertex list of the polygon so that the parts of the polygon outside the trench rectangle are not drawn.

Computer In this document, the only computers discussed are IBM PS/2s and compatible machines. The SYGRAF program actually requires an IBM PC, PC-AT or PS/2 compatible computer with EGA or VGA graphics card, a Microsoft compatible mouse and a hard disk. In reality, however, an 80286-based computer with a hard disk faster than about 40ms is essential to prevent brain death from sheer boredom.

Context Any archaeological object, such as a pit, wall, post-hole, ditch, bank and so on which can be approximated in plan view as a polygon, and is not a find or a layer.

Database Generally refers to the site database, (eg 'the database'). This comprises fourteen DBASE .dbf files or tables and three CLIPPER index .ntx files.

MS-DOS The Microsoft MS-DOS operating system, or IBM PC-DOS as supplied with ibm computers. The program has only been used with DOS 3.2 and above.

dBase Ashton-Tate's DBASE III, DBASE III+ or DBASE IV relational database package. Also used as an adjective to describe things (especially files) which are directly compatible with DBASE. A CLIPPER database file, for example, may be described as a DBASE file for convenience.

Find Any small archaeological object, such as a bone, pot sherd, coin, mussel shell and so on, which is not a context and is considered to have been found at a point of the site.

Index Refers to DBASE or CLIPPER index files, which are suffixed with .ntx (CLIPPER) or .ndx (DBASE).

Laserdisk See **Videodisk**

Layer A unit of the site which may contain finds and contexts and which the program treats as one unit. Layers are numbered from 1 (considered to be at the top) to as much as 9 (considered to be the bottom).

Site or Site database See **Database**.

Table Refers to one table of either the site database or the user's database. In DBASE, tables are equivalent to files and are suffixed in DOS by .dbf to distinguish them from other files such as .ntx and .ndx (index) files.

Trench Refers to an excavated area **of one layer**. Thus to create a hole through four layers, four 'trenches' are required, one in each layer, and each inside the other.

User The student using the system, assumed to have minimal knowledge of both computers and databases.

User's database The subset of data which represents the user's site, stored in his subdirectory of the site directory.

Videodisk This refers to a CD rom disk and associated disk drive, particularly the Philips VP 410 Laservision Disc Drive.