COMPASS & TAPE

Volume 8 *Number 1* *Summer 1990*

Compass & Tape is the quarterly newsletter of the Survey & Cartography Section of the National Speleological Society, Cave Ave., Huntsville, Alabama 35810, U.S.A. The cost is $4.00 per year for four issues. Please make checks payable to: Survey & Cartography Section. Include your NSS number for Section membership. Foreign members and subscribers are welcome! Rates are US$4.00 per year for surface mail: inquire for air rates. We regret that payment must be in US$, and checks drawn on U.S. banks. The volume runs from the annual NSS Convention: those paying later will receive all back issues. Expiration dates are printed on mailing labels. Volumes 1, 2, 3, and 4 are available for $5.00 each.

Send Dues, Subscriptions,
Address Changes to:
Rich Breisch

Send Articles, Photos, Letters,
Comments, Tips, Maps to:
Tom Kaye

Cover: Drawing of a cave surveyor by Linda Heslop. (Thrice expanded by xerography.)

Printed by members of the DC Grotto and the Potomac Speleological Club

MISCELLANEOUS OPERATIONS FOR STAGE-4 CAVE MAPS

by

Fred L. Wefer

1. INTRODUCTION

In recent papers in this journal, Wefer (1989a, 1989b, 1989c, and 1990a) introduced the concept of "stages" in the development of the computerization of cave mapping, presented the design of a 3D north/scale icon, discussed viewing definition and control, and discussed content definition and control for Stage-4 cave maps.

A Stage-4 cave map is a map designed to be viewed on the computer graphics screen. The information content of the cave map is conveyed via lines, symbols, text, and polygons comprised of pixels on the screen. Extensive use is made of color. The content of the map can be changed at the option of the viewer. Any portion of the cave may be viewed in any 3D direction at any reasonable scale, all at the option of the viewer. Sequences of changes in both viewing and content can be defined interactively by the viewer and played back in a movie-like fashion. The hardware and software which make all this possible are integral parts of the map.

A computer program now called Interactive Cave Map (ICM) was used by Wefer et al (1983) to illustrate the application of interactive computer graphics to cave mapping. ICM is written in FORTRAN and makes extensive use of a Commercial Off The Shelf (COTS) software product called TEMPLATE (a graphics package based on the proposed CORE graphics standard). ICM has been used by this author over the past eight years as a prototype for Stage-4 cave mapping.

Previous papers in this series have discussed two types of operations into which most ICM functions fall. The viewing operations (discussed in Wefer (1989c)) were implemented in ICM via twenty-four commands and corresponding menu options. The content control operations (discussed in Wefer (1990a)) were implemented in ICM via forty-two commands and corresponding menu options.

This paper discusses a set of miscellaneous, though important, operations available for Stage-4 cave maps. It is assumed that the reader is familiar with the previous papers in this series. The north/scale icon discussed by Wefer (1989b) and the viewing and content options discussed by Wefer (1989c and 1990a) were used in creating the illustrative examples of this paper.

2. HIGH PERFORMANCE GRAPHICS

This paper begins to move into areas of computer graphics that may not be familiar to the caver who typically uses a Personal Computer (PC) to reduce cave survey data. In order to make really effective use of some of the miscellaneous operations discussed below, a high performance graphics device is required. The reader may never have seen such a device, hence it is appropriate to say a few words about these devices: what they can do, what they are, and how they work.

2.1 HIGH PERFORMANCE GRAPHICS DEVICES

By high performance graphics device is meant a computer graphics terminal or workstation capable of at least the following:

- Displaying at least 16 user defined colors simultaneously,
- Having a screen resolution of approximately 1000x1000 pixels,
- Accepting coordinates of graphics primitives in 3D,
- Handling user defined sets of graphics primitives as logical units,
- Performing dynamic viewing operations on such logical units,
- Changing attributes of such logical units.

A high performance graphics device is capable of displaying at least sixteen colors on the screen simultaneously, and to do this via a Color Lookup Table (CLT). This allows the application program to specify a color in terms of a color index that is translated in the hardware, at the last stage of processing, into the appropriate mixture of red, green, and blue (RGB). Colors can be changed almost instantaneously by changing the translation specified in the CLT. While color is expensive and difficult to use in publications such as this, it is very effective in the display of map information on the screen.

A high performance graphics device has a resolution on the screen of approximately 1000x1000 pixels (picture elements). This is about twice the resolution of a standard U.S. television monitor, about the same resolution as a high-definition television monitor. This resolution is required for the crisp lines necessary to portray the detail of Stage-4 cave maps.

A high performance graphics device is capable of accepting coordinates of graphics primitives (lines, symbols, text, and polygons) in 3D. It is possible, of course, to develop a cave map computer program in 2D only, and hardware is available that handles only 2D coordinates. In fact, one of the graphics standards, the Graphical Kernel System (GKS), is 2D only. This is adequate for many Computer Aided Design (CAD) applications (e.g., Stage-3 cave maps); however, it is generally inadequate for Stage-4 cave maps.

A high performance graphics device is capable of handling user defined sets of graphics primitives as logical units of the picture displayed. This allows the application program to define and then work with entities comprised of collections of primitives, e.g., passage walls made up of collections of polygons, or map symbols made up of collections of lines. These correspond to the content features discussed in Wefer (1990a).

A high performance graphics device is capable of performing dynamic viewing operations on such logical units. Such operations include translating, scaling, and rotating the logical unit on the screen in 3D, almost instantaneously. By almost instantaneously is meant in less than a tenth of a second for a logical unit comprised of tens of thousands of primitives.

A high performance graphics device is capable of changing attributes of such logical units. Visibility, the most important such attribute, can be changed almost instantaneously without unintentionally affecting the attributes of other logical units on the screen, even temporarily.

Some examples of high performance graphics devices from the past and the present are: Evans & Sutherland MPS and PS300; Megatek 3355 and 7255; Tektronix 4115 and 4129; IBM 5080; Silicon Graphics IRIS; and the IBM RISC System/6000.

2.2 DISPLAY LIST MACHINES

Display list machines are a class of high performance graphics devices with the above described capabilities. Figure 1 below shows a very simplified schematic diagram illustrating how a display list machine works, in conjunction with a COTS graphics package like TEMPLATE running on a host computer. The major components and their functions are as follows:

- Input/Output Processor (IOP) - receives graphics instructions from the host and stores them in the HDL, updating logical units as necessary. The IOP also sends information back to the Host.
- Hardware Display List (HDL) - memory for the storage of logical units comprised of lines, symbols, text, and polygons. The information in the HDL is essentially in the form of instructions for the DLP.
- Display List Processor (DLP) - cyclically reads the HDL and on each cycle draws into the bit planes the visible logical units, applying a 3D image transformation to the 3D coordinates of all graphics primitives within each visible logical unit. The DLP runs fast enough to update the bit planes in about 0.1 seconds, even if the HDL contains tens of thousands of graphics primitives.
- Bit Planes - two sets of bit planes for the approximately one million pixels on the screen. While the DLP is drawing in one set, the DAC is refreshing the screen from the other. The bit planes essentially contain a color index for each pixel on the screen.
- Color Lookup Table (CLT) - a table of RGB values used by the DAC in converting the color indices in the bit planes into colored lines on the screen.
- Digital to Analog Converter (DAC) - converts the color index stored for each pixel in the bit planes (using the CLT) and refreshes the screen. The DAC runs fast enough to refresh the screen at something like 45 Hz.
- Screen - a standard (today) 1000x1000 color raster Cathode Ray Tube (CRT) computer graphics screen.
- Keyboard and Mouse - input devices attached to the display device.
- Peripheral Control Unit (PCU) - controls input devices such as the keyboard and the mouse, and passes information from them to the IOP.

- Software Display List (SDL) - memory in the host processor used by TEMPLATE to maintain a device independent replica of the HDL.

- Host - the computer or workstation on which the application program (ICM) and the graphics package (TEMPLATE) are running.

Note: Every display device manufacturer has its own architecture, with special names for the components and slightly different functions. The schematic in Figure 1 is intended to present a very simplified (no really) view of how these rather complicated machines work. In the case of a graphics workstation the Host is conceptually contained within the graphics device (or vice versa).

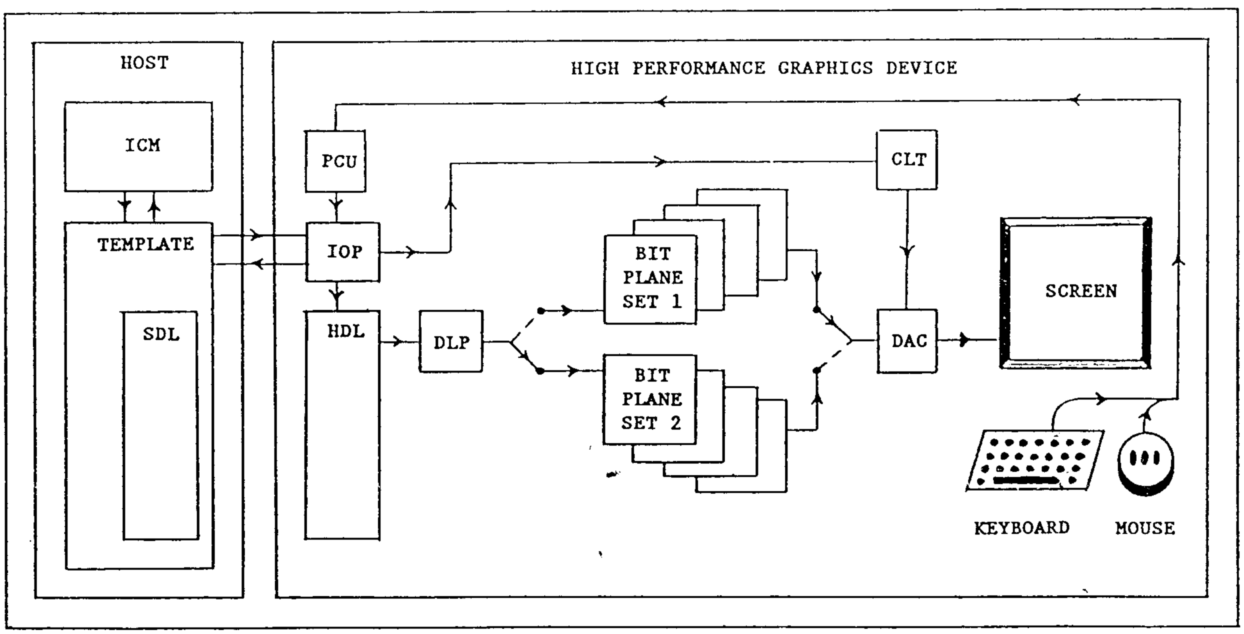

Figure 1. Schematic diagram of a display list machine, a class of high performance graphics devices useful for Stage-4 cave maps.

2.3 VIEWING OPERATIONS

So, how does all this work? To begin with, remember that even if the caver is just sitting there looking at the screen, a lot is still happening inside the box. The DAC is refreshing the CRT at about 45 Hz. The DLP is reading the HDL and drawing all visible logical units (including the cave) into the set of bit planes not being used by the DAC. The DLP requires about 0.1 seconds to complete one cycle through the HDL, after which the DAC and DLP simultaneously switch bit plane sets and the cycle starts over. During this process the caver sees nothing happen on the screen.

Now let's take the example of a simple viewing change as discussed in Wefer (1989c). Suppose the caver places the mouse cursor on the "+" sign next to the "A" in the "10.0" row of the viewing control menu (see Figure 4 of Wefer

(1989c)). The PCU tracks the position of the cursor and sends this information to the IOP which updates the cursor location in the HDL. The cursor that you see on the screen is just a special logical unit in the HDL, processed by the DLP and the DAC like all the rest.

When the mouse button is pushed, this action is signaled to the host by the PCU through the IOP. The host gets the current coordinates on the screen of the mouse cursor and also which button was pushed. TEMPLATE looks up in the SDL what menu item the caver has picked and passes this information on to ICM.

Note: Instead of sending back to the host the coordinates of the cursor, some display list machines do this "picking" operation in the HDL via still another processor, and send back to the host the identification of the item picked.

In response ICM increments its azimuth variable by 10.0 degrees and tells TEMPLATE to update the display. TEMPLATE builds a new 3D image transformation (actually a 4x4 matrix), updates the matrix in the cave logical unit in the SDL, and sends the matrix to the IOP. The IOP updates the matrix in the cave logical unit in the HDL.

On the next cycle the DLP reads the new matrix, applies it to the primitives in the cave logical unit, and draws the resulting transformed primitives into the set of bit planes. At the end of the cycle the DAC and DLP simultaneously switch bit plane sets. When the DAC next refreshes the screen the cave appears in the new orientation on the screen.

The miracle is that the elapsed time, from the press of the mouse button to the cave appearing in its new orientation on the screen, is only about 0.1 seconds. The PC you use to reduce your cave survey data does not (yet) have an HDL or a DLP. It has one set of bit planes (usually) into which the PC equivalent of the IOP directly writes graphics information, and a DAC that refreshes the screen.

2.4 DYNAMIC VIEWING OPERATIONS

By <u>dynamic viewing operation</u> is meant an operation that involves changing the viewing in such a fashion that the cave appears to be smoothly moving on the screen. Suppose you want to perform a simple dynamic viewing operation like spinning the cave in azimuth. Program ICM simply executes a loop like this:

(a) ICM increases the azimuth variable by, say 2 degrees,
(b) TEMPLATE constructs the 4x4 image transformation matrix,
(c) TEMPLATE updates the SDL on the host,
(d) TEMPLATE sends the new matrix to the IOP,
(e) IOP updates the HDL with the new matrix,
(f) DLP reads the HDL and draws the visible logical units (including the cave transformed by its new matrix) into the bit planes,
(g) DLP and DAC simultaneously switch bit plane sets,
(h) DAC converts the bit planes (using the CLT) and refreshes the screen,
(i) if the azimuth has not increased by a total of 360 degrees, go to (a), else
(j) exit the loop.

What the caver sees on the screen is a 360 degree spin of the cave. I state without proof that sometimes you can learn more about a cave by watching one spin of a Stage-4 cave map than you can learn by studying a traditional map of the same cave for a long time.

2.5 CONTENT CONTROL OPERATIONS

Content control operations were extensively discussed in Wefer (1990a). Toggling "off" a content feature like the screen grid is accomplished in a display list machine as follows:

(a) ICM changes its screen grid visibility flag to "off",
(b) ICM tells TEMPLATE to turn off the screen grid logical unit,
(c) TEMPLATE changes the visibility attribute of the screen grid logical unit in the SDL,
(d) TEMPLATE sends the same instruction to the IOP,
(e) IOP updates the HDL with the new visibility attribute setting,
(f) DLP (on the next cycle) reads the HDL and skips the screen grid logical unit, drawing all visible logical units (applying their image transformation matrices) into the bit planes, and
(g) DAC converts the bit planes (using the CLT) and refreshes the screen.

A tenth of a second after the menu option "SG" is picked, the screen grid is missing from the screen. Everything else appears just as before, i.e., unchanged.

At a very high conceptual level, that is how a high performance graphics device works. If you run ICM on other than such a device, TEMPLATE does the best it can using whatever capabilities the device has. The figures in this paper were made by running ICM on a SUN 3/260, which is not a high performance graphics device. Now, let's move on to the real subject of this paper, namely miscellaneous operations for Stage-4 cave maps.

3. BACKGROUND INFORMATION

There exists a set of operations for Stage-4 cave maps that fall outside the bounds of the two sets of operations previously discussed. These miscellaneous operations provide some very important capabilities in the areas of interface control, generation of hardcopies, special processing, dynamic viewing operations, etc.

Because these capabilities are well beyond the functionality of both the hardware and the software currently normally employed for cave mapping work, very little has appeared in the caving literature concerning them. Wefer et al (1983) demonstrated some of these capabilities in their video tape. According to Hoke (1983), Paul Hill's presentation at the 1983 NSS Convention must also have contained some information about these capabilities; however, I am unaware of any written published information on Paul's work. The material of Wefer et al (1983) is covered in later sections of this paper.

4. STAGE-4 REQUIREMENTS

Four considerations make necessary the provision of some operations beyond viewing control and content control, viz: (1) multiple interfaces are to be provided, (2) the cave map is both transparent and three-dimensional, (3) hardcopies are a necessity, and (4) maximum flexibility is desired.

Previous papers in the series have discussed two user interfaces, a Command Language Interface (CLI) and a Graphics Menu Interface (GMI). Some operations are more efficiently performed with the GMI, some are more efficient with the CLI. The following example illustrates the point.

Suppose the ICM cave map is being viewed in azimuth 57.5 degrees and the caver wants to change this to 100.0 degrees. This can be accomplished on a SUN 3/260 via the GMI by the following sequence of steps:

(a) Select the "+" next to the "A" in the "1.0" row via mouse button number two (the azimuth is increased to 59.5),
(b) Select the "+" next to the "A" in the 0.1" row via mouse button number three (the azimuth is increased to 59.8),
(c) Select the same thing again via mouse button number two (the azimuth is increased to 60.0),
(d) Select the "+" next to the "A" in the "30.0" row via mouse button number one (the azimuth is increased to 90.0), and finally
(e) Select the "+" next to the "A" in the "10.0" row via mouse button number one (the azimuth is increased to 100.0).

If the CLI were being used, the same thing could be accomplished by:

(a) Enter "A=100.0" (azimuth is set to 100.0).

In general, the GMI is more efficient when exploring by trial and error. The CLI is more efficient when you know exactly where you want to go. An experienced viewer makes use of both interfaces; therefore, a mechanism is needed to quickly and easily switch from one interface to the other and back again.

The 3D nature of Stage-4 cave maps often results in viewing situations where one passage lies in front of another. Since traverse lines and passage walls are both normally transparent, it is difficult to tell which passage is in front of which. Some provision needs to be made to resolve these ambiguities.

Until recently, Stage-4 cave maps were limited to displaying traverse lines information only. The inclusion of wire-frame models of passage walls, and symbols within the volume defined by those passage walls, necessitates that the caver be given some special processing options in order to take full advantage of the passage walls and symbols information.

Some operations are required in "other" areas. For example, while Stage-4 cave maps are designed to be viewed on the computer graphics screen, some provision must be made for publishing results and sharing information with cavers who cannot personally view the cave map on the screen. Hardcopy functions need to be provided. The figures for the papers in this series would not have been possible without this functionality. Still other options are required to maximize the flexibility of the Stage-4 cave map. For example, the viewer needs the capability of changing some of the parameters of the map while it is being viewed (colors, north/scale positions, grid plane locations, etc). An option is even needed for exiting the program.

The above discussed considerations result in requirements to provide miscellaneous operations for Stage-4 cave maps in the following four areas:

- Interface control,
- Dynamic viewing operations,
- Special processing, and
- Other.

Exactly how the above discussed miscellaneous operations are provided is the decision of the system designer. I continue to stress that more than one solution exists. The reason the requirements are presented in a separate section is to provide the system designer a view of what the requirements are, unbiased by the approach to satisfying them that I used in program ICM.

5. STAGE-4 DESIGN ELEMENTS

A list of design elements which, by experimentation via program ICM, have been found to satisfy the above discussed requirements is shown below.

- Interface control,
    * A CLI command for switching to the GMI, and
    * A GMI option for switching to the CLI,
- Dynamic viewing operations,
    * Rotations in azimuth (spin),
    * Oscillations in azimuth (yaw),
    * Oscillations in dip (pitch), and
    * Oscillations in bank (roll),
- Special processing,
    * Hidden line removal processing,
    * Cutaway view processing, and
- Other,
    * Making hardcopies,
    * Changing run parameters,
    * Resetting to the initial viewing state,
    * Resetting to the initial content state,
    * Displaying the current content state, and
    * Exiting the program.

The two user interfaces (GMI and CLI) that were used for viewing control and content control are used in a consistent manner to control the miscellaneous operations. In the particular case of interface control, a GMI option is provided to switch to the CLI, and a CLI command is provided to switch to the GMI. Of course, no GMI option is required to switch to the GMI. No CLI command is required to switch to the CLI, although typing the command "CLI" is not treated as an error by ICM. It does nothing, but it is very fast!

Sometimes all that is needed to remove ambiguities in passage location is to move the cave a little. If this motion is of a "predictable fashion", then which passage is in front will be readily apparent. Accordingly dynamic viewing operations are provided for this purpose. Experimentation via ICM has established four "predictable fashions" useful for this purpose: rotating the cave in azimuth and oscillating the cave in the three viewing angles, azimuth dip, and bank.

Having a replica of the HDL accessible to the COTS graphics package (as discussed in Section 2) means that special processing can be performed by the software that is not available in hardware even with high performance graphics devices. The two types of special processing needed are hidden line removal and cutaway view.

Passage walls can be represented in ICM by triangular polygons forming polyhedra. Since these are transparent, one normally sees both the passage wall nearest the viewer and also the wall on the other side of the passage. Dynamic viewing operations help to remove ambiguities, but the resulting display is still confusing. The solution to this problem is hidden line removal, i.e., for the current viewing and content state, erase the screen and redraw the cave leaving out lines which would not be visible were the cave a solid object (such as a clay model). This functionality was added to TEMPLATE in Version 6.0 and is available via a single FORTRAN subroutine call.

Following the incorporation of hidden line removal in ICM, it became clear that something more was needed. Symbols information (cave map symbols) is placed within the volume defined by the passage walls information. The application of hidden line removal to remove the ambiguities in passage walls information completely removes the symbols information from the screen!

Hidden line removal is a technique for representing the outsides of wire-frame models. What was needed was a technique for representing the insides of wire-frame models. A technique called cutaway view has been developed by Wefer (1990b) especially for this purpose.

The ICM design elements included under the heading other include: resetting to the initial viewing state, resetting to the initial content state, displaying the current content state, making hard-copies, changing run parameters, and exiting the program. Some of these have been briefly discussed in previous papers in this series. They are included again here for completeness.

6. THE ICM IMPLEMENTATION OF MISCELLANEOUS OPERATIONS

Sixteen CLI commands and fifteen GMI options are used to provide the miscellaneous operations discussed above. Except for the case of interface control operations already mentioned, all of them are available via both interfaces.

6.1 THE ICM GMI OPTIONS FOR MISCELLANEOUS OPERATIONS

The ICM function menu containing the fifteen GMI options for miscellaneous operations is shown in Figure 2 below. It appears along the lower-left-hand edge of the screen. The two-character commands in this menu are placed three in a row and separated by "/"s. The user selects an option by placing the pick device cursor on either of the two characters and pressing the button. If the user places the cursor on a "/" and presses the button, nothing happens (or at user option a message is displayed in the alphanumeric window, just above the device viewport at the top of the screen on the SUN 3/260).

EX

Note: The script menu goes in here.

```
RC/SC
SP/RP/CS
SU/CF/ML
YW/PT/RL
MU/HL/CV
HP/NS/EW
FM/SG/VB
TO/NC/NP
TL/PW/SY
N1/N2/N3
N4/N5/N6
S1/S2/S3
S4/S5/S6
P1/P2/P3
P4/P5/P6
T1/T2/T3
T4/T5/T6
RV/CP/CM
TB/PS/CL
```

Figure 2. The ICM function menu is shown. It contains the fifteen GMI options for miscellaneous operations. The two-character options in this menu are separated by "/"s. The user selects an option by placing the pick device cursor on either of the two characters and pressing the button. The option "EX" is placed near the top of the screen away from the other options so that is is less likely to be picked by accident.

6.2 THE ICM CLI COMMANDS FOR MISCELLANEOUS OPERATIONS

In Table I below the sixteen commands for the miscellaneous operations are divided into four groups: (1) interface control, (2) dynamic viewing operations, (3) special processing, and (4) other. Each of these is discussed below and illustrated via hardcopies of Stage-4 cave maps of the following caves: Corkscrew Cave, the mythical cave used in previous papers of this series; The Butler Cave-Sinking Creek System, a large cave system in West-Central Virginia; and Cueva Catanamatias, a vertical cave located in the San Juan Province of the Dominican Republic.

TABLE I. The sixteen ICM miscellaneous operations commands are listed. The CLI command is shown followed by the command meaning. Underlined and capitalized characters in the meaning indicate the origin of the command name.

COMMAND	MEANING
-------	INTERFACE CONTROL OPERATIONS
CL	switch to the Command Language interface.
GM	switch to the Graphics Menu interface.
-------	DYNAMIC VIEWING OPERATIONS
PT	PiTch the cave (oscillate the cave in dip).
RL	RoLl the cave (oscillate the cave in bank).
SP	SPin the cave (rotate the cave in azimuth).
YW	YaW the cave (oscillate the cave in azimuth).
-------	SPECIAL PROCESSING OPERATIONS
CV	construct a Cutaway View.
HL	remove Hidden Lines from the screen.
PS	PoSt the segments to the device.
-------	OTHER OPERATIONS
CM	Copy the screen with graphics Menus.
CP	CoPy the screen without graphics menus.
CS	show Current Status.
EX	EXit program ICM.
RC	Reset to the initial Content state.
RP	change Run Parameters.
RV	Reset to the initial Viewing state.

6.3 INTERFACE CONTROL OPERATIONS

Interface control operations are simple to use. If you are using the CLI and you want to switch to the GMI, simply type the command "GM" at the CLI prompt. Note that pressing the mouse button while using the CLI has no effect.

If you are using the GMI and you want to switch to the CLI, simply select option "CL" in the ICM function menu. Note that the function menu and the viewing menu remain visible even though you are using the CLI. If you wish them to disappear, type the command "MU" to toggle them "off". Actually, it may be useful to leave them "on" as a reminder of what commands are available.

6.4 DYNAMIC VIEWING OPERATIONS

Dynamic viewing operations are most useful in resolving ambiguities in maps containing only traverse lines information. There are four dynamic viewing operations in ICM: <u>spin</u>, <u>yaw</u>, <u>pitch</u>, and <u>roll</u>. Each is illustrated below.

Note: These operations cannot readily be demonstrated in figures on paper. The illustrations below use multiple images of the cave to give you an idea of what the operations accomplish. ICM was slightly modified in order to produce the multiple images for these figures. ICM doesn't normally work quite that way. To fully appreciate the power of these techniques, you need to look at Stage-4 cave maps on a high performance graphics device.

Rotations in azimuth (called <u>spin</u>) are initiated via command "SP". The spin step size (angular increment between images) and the number of spin rotations to be performed are both run parameters.

Note: At a point early in the construction of the map, parameters specifying details of the map are input from a disk file. These parameters are of two types, run parameters and setup parameters. <u>Run parameters</u> can be changed interactively while viewing the Stage-4 map via command "RP" (see the discussion below). <u>Setup parameters</u> cannot be changed after the map is constructed.

Figure 3 below shows Corkscrew Cave before and after the user has selected "SP". The spin step size used in Figure 3 is 10 degrees, hence 36 images of the cave viewed in different azimuths are shown.

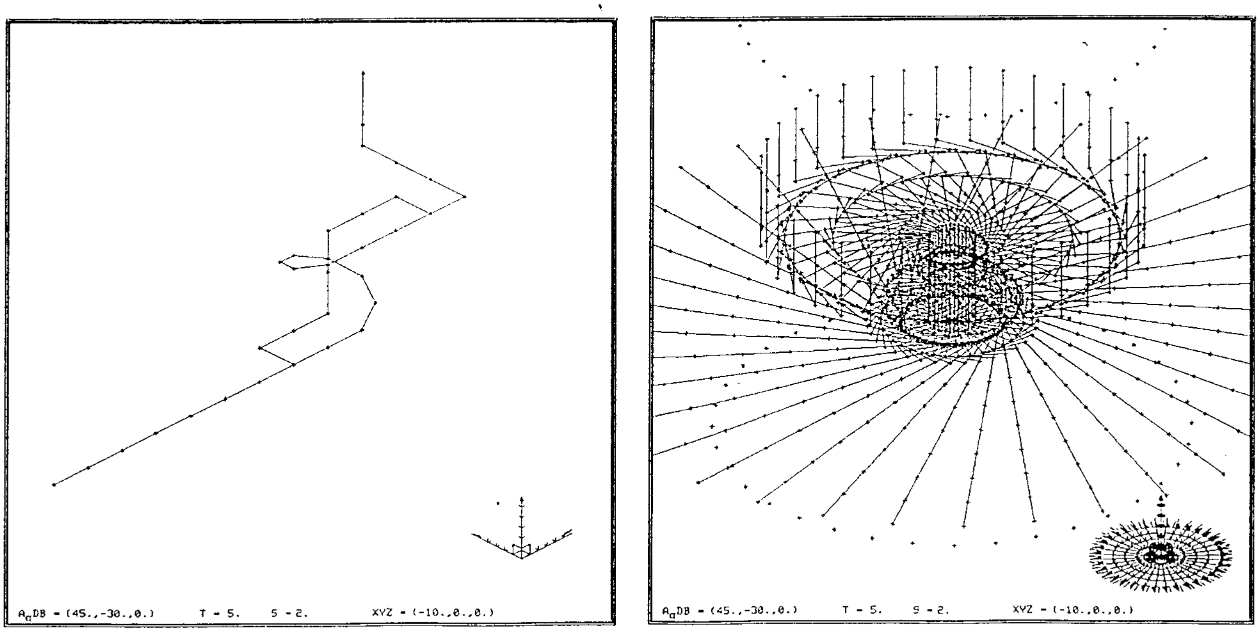

Figure 3. Corkscrew Cave traverse lines information only, before and after option "SP" is selected.

If you were looking at this operation on the screen of a high performance graphics device, only one of the images would be visible at any moment in time. What you would see is the cave spinning in 3D, as if you were watching a movie. Actually, the spin would be rather fast. A smaller spin increment would generate more images per rotation and a more comfortable spin rate.

In some cases a full 360 degree spin is not necessary, a short oscillation being sufficient to resolve the ambiguities. Oscillations in azimuth (called yaw) are initiated via command "YW". Figure 4 shows a portion of Corkscrew Cave before and after the user has selected "YW". The yaw step size, the range of yaw (amplitude of the oscillation), and the number of cycles (oscillations) are run parameters. The values used in Figure 4 are 2.5 degrees, 10 degrees, and 1 cycle, respectively.

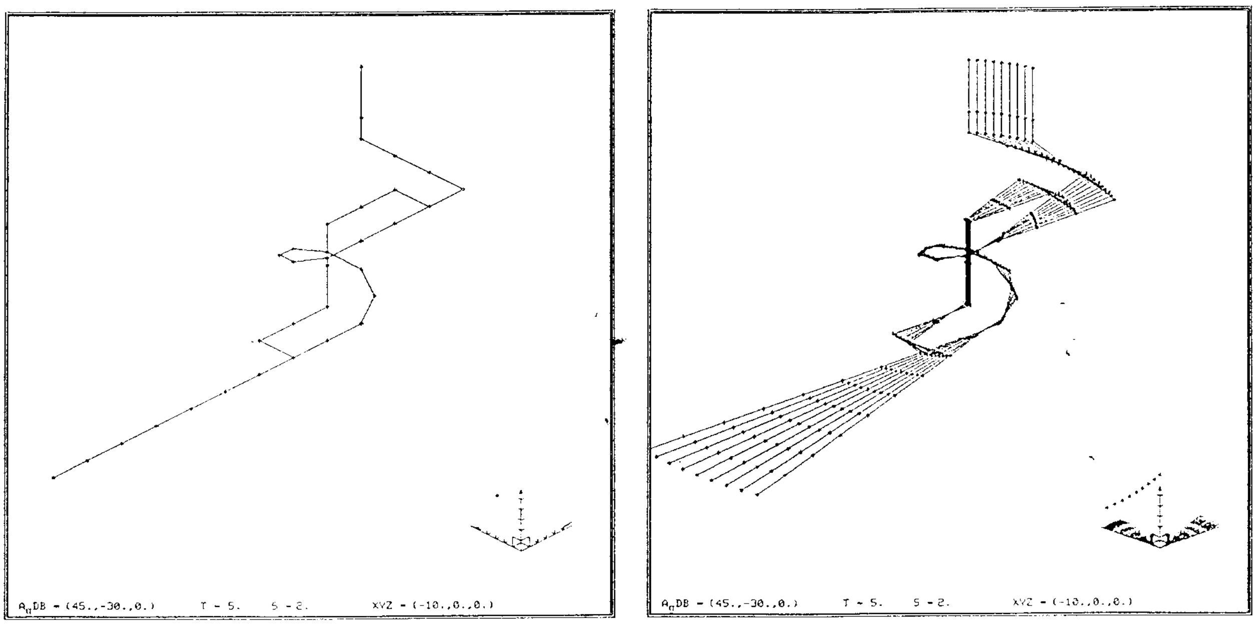

Figure 4. Corkscrew Cave traverse lines information only, before and after option "YW" is selected.

Oscillations in dip (called pitch) are initiated via command "PT". Figure 5 below shows a portion of the Butler Cave-Sinking Creek System before and after the user has selected "PT". The pitch step size, the range of pitch, and the number of cycles are run parameters. The values used in Figure 5 are 3 degrees, 9 degrees, and 1 oscillation, respectively.

Oscillations in bank (called roll) are initiated via command "RL". Figure 6 below shows a portion of Cueva Catanamatias before and after the user has selected "RL". The roll step size, the range of roll, and the number of cycles are run parameters. The values used in Figure 6 are 2.5 degrees, 10 degrees, and 1 oscillation, respectively.

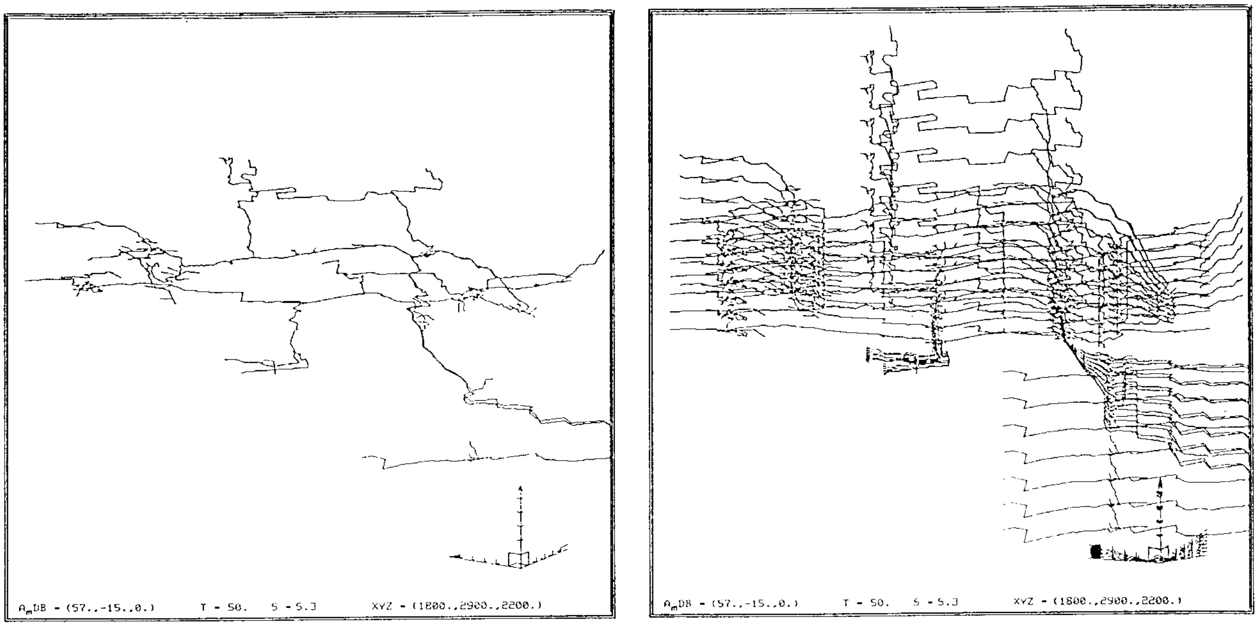

Figure 5. The Butler Cave-Sinking Creek System traverse lines information only, before and after option "PT" is selected.

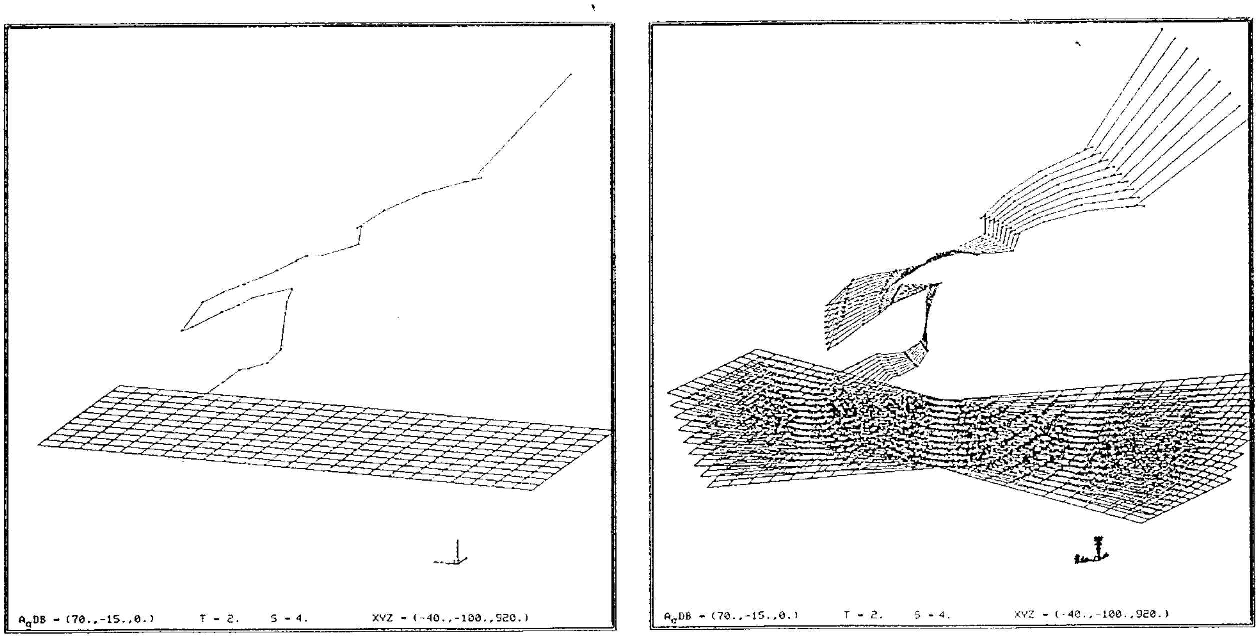

Figure 6. Cueva Catanamatias traverse lines information (plus the horizontal plane), before and after option "RL" is selected.

6.5 SPECIAL PROCESSING OPERATIONS

If passage walls information is being displayed, the dynamic viewing operations discussed above may not be adequate to resolve all ambiguities. Because TEMPLATE has a replica of the HDL in its SDL, special processing can be applied to the information making up the scene on the screen. Two special processing operations are available, <u>hidden line removal</u> and <u>cutaway view</u>.

<u>Hidden line removal</u> is initiated via command "HL". It causes the cave to be erased and redrawn with the elimination (removal) of all lines (and portions of lines) and all polygons (and portions of polygons) which would not be visible were the passage walls opaque. Figure 7 below shows a portion of Cueva Catanamatias before and after the user has selected "HL". Note that the symbols information (pool of water, standing caver, prusiking caver, rope, and survey stations) visible before "HL" was selected is missing after hidden line removal is applied.

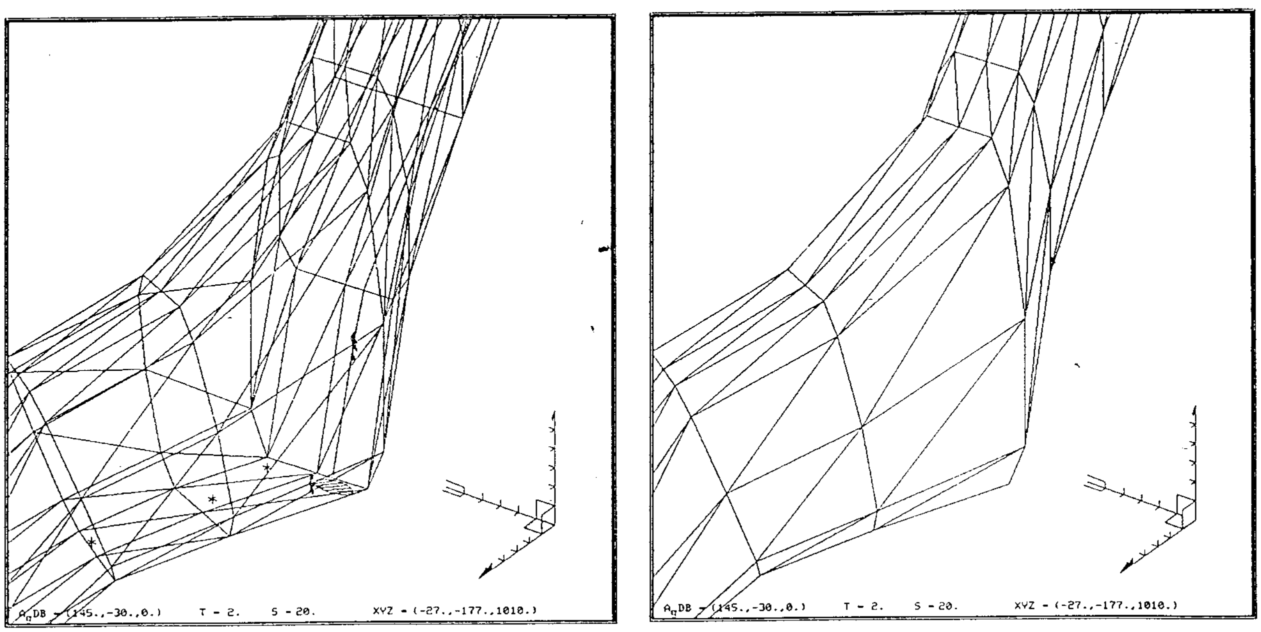

Figure 7. Cueva Catanamatias before and after option "HL" is selected.

<u>Cutaway view</u> is initiated via command "CV". Its operation is a little harder to explain (see Section 8 below). It shows you not the outsides of the passages but the insides of the passages. Figure 8 below shows Cueva Catanamatias before and after the user has selected "CV". Note that the symbols information, missing when "HL" was applied, is now properly shown.

When either "HL" or "CV" processing is shown on the screen, this fact is reflected in the current content state display initiated via command "CS" (see Figure 10 of Wefer (1990a) for an example).

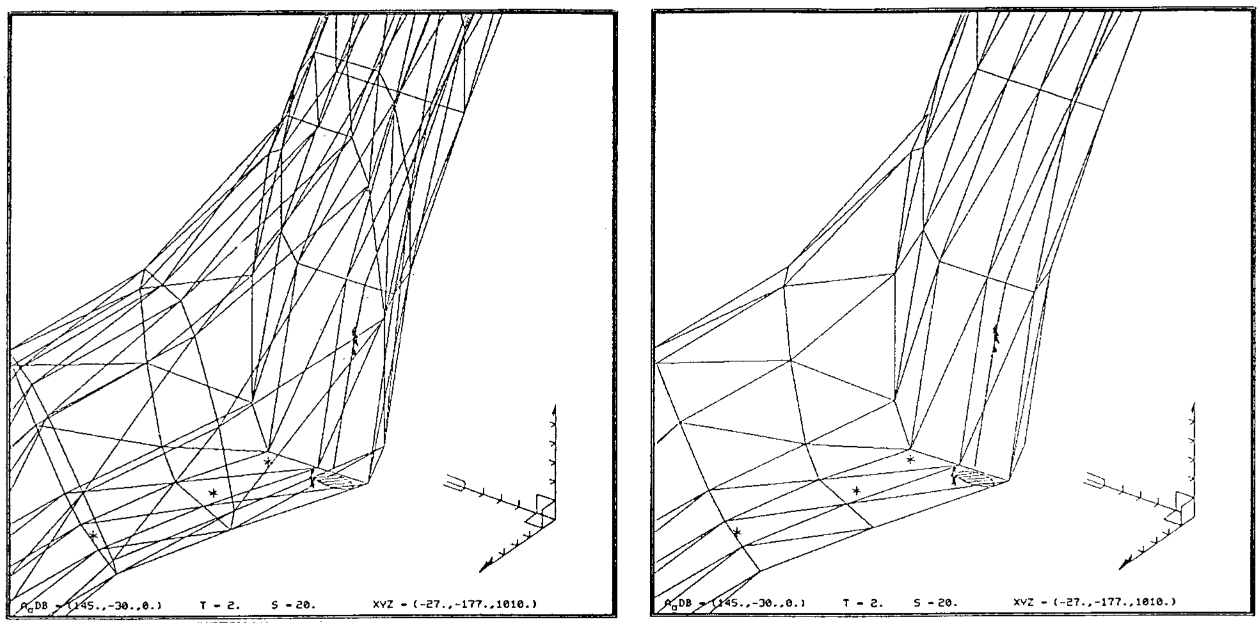

Figure 8. Cueva Catanamatias before and after option "CV" is selected.

Any subsequent viewing or content change causes the special processing display to be erased and the normal wire-frame display to reappear. To erase the special processing display without changing the viewing or the content, an operation called posting is provided. Simply pick option "PS".

6.6 OTHER OPERATIONS

Many high performance graphics devices allow a hardcopy to be made under the control of the application program. For such devices, ICM provides two commands. Initiating command "CP" causes the screen to be copied without the graphics menus, i.e., the menus are toggled off (as if "MU" had been typed), the hardcopy is made, and the menus are toggled back on.

Initiating command "CM" causes the screen to be copied as is. If the menus are "on", they are included in the hardcopy.

If the device does not support this functionality, these commands merely cause the audible alarm (bell) to sound and a message to that effect to be displayed in the alphanumeric window of the screen.

The current content state can be displayed via command "CS". This causes ICM to display, in the alphanumeric window, the current state of all content feature toggle switches, special processing operations ("HL" and "CV"), and whether or not the GMI is being used ("GM") (see Figure 13 of Wefer 1990a).

The map can be reset to the initial viewing state by typing command "RV". The initial viewing state variables are setup parameters. The map can be reset to the initial content state by typing command "RC". The initial content state is also a setup parameter. Because these operations can make significant (and not easy to undo) changes in the map, the caver is asked to verify that he wants to do these operations by typing "y".

A number of parameters can be changed while the caver is viewing the Stage-4 cave map. These run parameters are changed via a simple alphanumeric menu initiated via command "RP". The run parameters include the following:

```
PERFORMANCE
   BEEP WHEN ICM IS READY FOR NEXT INPUT (off/on)
   ECHO THE I.D. OF THE OPTION PICKED (off/on)
   MENU SCALE FACTOR (0.0 - 2.0)
   PICK BOX WIDTH (inches on the screen)
   PICK BOX HEIGHT (inches on the screen)
   MOVIE ABORT FLAG (no/yes)
   MOVIE SCRIPT STEPS FACTOR (0.0- 10.0)
   SAMPLING DELAY TIME (seconds)
SPIN PARAMETERS
   NUMBER OF SPIN ROTATIONS
   SPIN STEP SIZE (degrees)
PITCH PARAMETERS
   RANGE OF PITCH (degrees)
   PITCH STEP SIZE (degrees)
   NUMBER OF CYCLES
YAW PARAMETERS
   RANGE OF YAW (degrees)
   YAW STEP SIZE (degrees)
   NUMBER OF CYCLES
ROLL PARAMETERS
   RANGE OF ROLL (degrees)
   ROLL STEP SIZE (degrees)
   NUMBER OF CYCLES
NORTH/SCALE LOCATIONS
   IN CAVE NORTH/SCALE NUMBER (1-6)
      XYZ-COORDINATES (meters in the cave)
   PINNED NORTH/SCALE
      XYZ-COORDINATES (inches on the screen)
PLOT TITLE
   LINE NUMBER (1-5)
      TEXT STRING (76 characters/line)
      CHARACTER SIZE (Small, Medium, Large, or Extralarge)
      TEXT FONT (choice of 21)
      COLOR INDEX (0-15)
GRID PLANE COORDINATES
   X-COORDINATE OF NS-PLANE GRID (meters)
   Y-COORDINATE OF EW-PLANE GRID (meters)
   Z-COORDINATE OF  H-PLANE GRID (meters)
```

```
SETUP FILE
   NEW SETUP FILE NAME (40 characters)
COLOR INDICES
   COLOR INDEX TO BE CHANGED (0-15)
      NEW RED PERCENT (0-100%)
      NEW GREEN PERCENT (0-100%)
      NEW BLUE PERCENT (0-100%)
COLOR SET (0-3)
   STANDARD
   SPECIAL VIDEO
   USER DEFINED
   BLACK ON WHITE
```

Finally, when the caver types command "EX" she is asked to verify by typing "y" that she really wants to exit the program. If so, ICM terminates execution. If not, execution continues. The option "EX" is placed at the left edge of the screen near the top, away from the other options, to minimize the chance of the caver accidentally selecting it.

7. SUMMARY AND DISCUSSION

The miscellaneous operations provide some important capabilities for the viewer of a Stage-4 cave map. The four considerations of: multiple user interfaces being provided, maps being both 3D and transparent, hardcopies being a necessity, and the desire to maximize flexibility, all must be taken into account. The requirements for these miscellaneous operations for Stage-4 cave maps have been defined.

The ICM user interface provides access to these operations in a manner consistent with the viewing control and content control operations. Some of the miscellaneous operations (e.g., dynamic viewing operations) require a high performance graphics device for effective utilization, although they do execute on other devices, such as the SUN 3/260 used to make the above figures.

Another set of operations called <u>script operations</u> are available in ICM; however, the discussion of these is postponed to a later paper in the series. The reader is encouraged to think about other useful options for Stage-4 cave maps and to present them in future issues of this journal. The next paper of this series discusses the process of constructing content features containing traverse lines information for Stage-4 cave maps.

8. SOME TECHNICAL NOTES

This section is included for readers familiar with the terminology and technical details of CORE. The cutaway view functionality was developed by the author especially for ICM and reported in Wefer (1990b). It works as follows. During the construction phase of ICM initialization, the passage walls triangular polygons are all carefully defined with their normals pointing inward, i.e., towards the inside of the passage. When "CV" is selected, ICM instructs TEMPLATE to perform the following processing:

(a) Make the cave invisible,
(b) Scan the cave segment in the SDL and eliminate from consideration all polygons with normals that have a negative w-component (normal pointing away from the viewer (this is front-face removal)), and
(c) Perform standard hidden line removal on what's left.

The result is that the front-facing facets of the passage walls are eliminated, but when one passage lies in front of another, the correct obscuration is computed. Polygons of some content features (e.g., grid planes, corner squares of the north/scale, etc.) must be drawn twice (with normals pointing both ways) for this to always work. See Wefer (1990b) for more details.

9. REFERENCES

Hoke, B. (1983), "Summary of Computer-Related Convention Papers", CAS Newsletter, No. 11, pp. 1-3, June 1983.

Wefer, F.L., Igoe, J.W., and Gillen, P.A. (1983), "An Application of Interactive Computer Graphics to the Study of Caves", NSS Bulletin, Vol. 45, No. 2 (Insert), April 1983.

Wefer, F.L. (1985), "A User Interface for the Manipulation of 3D Objects", Proceedings of the 2nd Annual TEMPLATE User Network Conference, New Orleans, LA, 25-27 February 1985, 25 pages.

Wefer, F.L. (1989a), "The Computerization of the Cave Map", Compass & Tape, Vol. 7, No. 1, pp. 3-14, Summer 1989.

Wefer, F.L. (1989b), "A North Arrow and Scale for Stage-4 Cave Maps", Compass & Tape, Vol. 7, No. 2, pp. 3-12, Fall 1989.

Wefer, F.L. (1989c), "Viewing Definition and Control for Stage-4 Cave Maps", Compass & Tape, Vol. 7, No. 3, pp. 3-19, Winter 1989-90.

Wefer, F.L. (1990a), "Content Definition and Control for Stage-4 Cave Maps", Compass & Tape, Vol. 7, No. 4, pp. 3-23, Spring 1990.

Wefer, F.L. (1990b), "The Ins and Outs of Hidden Line Removal", Proceedings of the 1990 Template Graphics Software User Network Conference, Dallas Texas, 08-10 August, 1990, 33 pages.

*/**|***

CAVE MAP LANGUAGE: YET ANOTHER CAVE COMPILER

by Mel Park, NSS 25729

YACC might have been a good name for this thing that I have done, but it's already taken (by a UNIX tool) and besides, I long ago picked the more mundane name of Cave Map Language.

In 1988, CRF, cooperating with the National Park Service, began reconsolidating its survey data of Mammoth Cave into a unified format. At that time we certainly didn't think it wise to write yet another data reduction program for cave survey data, particularly since Mammoth Cave National Park was going soon to contract Doug Dotson to write an all-encompassing Geographical Information System that would include cave data reduction. But as several of us, Jim Borden, Scott House, together with Tom Kaye, began to examine the size of the problem and the several special requirements of Mammoth Cave, a number of facts began to emerge and a niche for a new program and a new approach to representing data, designed specifically for the huge Mammoth Cave data base, became apparent. Mostly, we wanted a format for archiving the data on computer that would retain its utility through the successive generations of computers and mass storage devices that we knew would inevitably come. Secondly, we needed tools (programs) that could read the data and aid us in merging our over thirty years of survey data. When we started, the data from most of our 2500 survey books existed in four distinct formats and media. We had to have programs that could translate these different formats into the new one and then process the data and present it to us in various forms.

At the very beginning, I decided to adopt the idea that the data were to be encoded in a language, that is, not as just rows and columns of data, but rather within a set of syntactical rules that would permit writing down just about any conceivable aspect of a cave survey. The data had to be machine-readable, of course, so that it could be processed by computer, but I also wanted the data to be in a form that a human would find easy to read and, more importantly, to type. This meant ASCII files and a relatively free format. If the format could pretty much follow the practices cavers use in writing the data down in survey books, then it would have the advantage of seeming natural and not require a lot of explanation in order to be understood. So, for example, the language would allow redundant or optional fields to be simply left out, just as they are in a survey book. More subtly, there would be considerable freedom in the way that stations and survey shots would be matched to each other. That is, within a given survey, the station name appearing on the same line as the data for a particular shot could be the station from which the shot was made, or the one to which the shot was made, or both stations could appear on the line (compare Figures 1 and 6). After all, any caver can look at a set of data and, with a little head scratching, figure out the order of things. Well, so can a computer, provided that there is a reasonable set of rules. There would have to be adequate ways of inserting comments into the data and an ample supply of commands that would direct the way that the data are interpreted and processed by both the human and the computer.

Now, deciphering all of this requires a more complex computer program than is the norm in cave data reduction programs, with the possible exception of SMAPS, which is a full-featured integrated system. I felt that that was a reasonable price to pay for the added power and flexibility of the system.

```
;Thanksgiving Expedition, 1989
#openFSB 2463
#c Logsdon River
#d 11-24-89
#co 316822 0,180
#i 825374 +.5,0
#p D.Coons(B),J.Branstetter(Co),N.Pace(Ch),J.Fant(Ch)
;#c 641417 1,181 (Backup compass)
;#i 840020 0,1 (Backup Clino)
$190X24->K1: 25         89,270          -4,4        8,14,3,5
K1:          50.4       105,285.5       -1,1        20,8,3,2
             50.3       118.5,299       4,-4        3,12,3,2
             42.8       114,293.5       1,-1        3,5,0,2.5
             30.1       53,234          -2,2.5      8,12,4,3
             32.7       26.5,206        4,-3        18,3,9,3 (lead 1hx2h at K5 )
             50.7       103,283         1,1         8,9,0,3.5
             51.3       152.5,333.5     .5,-1       5,18,0,3
             50.6       115,295         0,-1        4,12,0,3
                                        (lead 2hx10w, upstream, some air, at K8)
K9:          44.6       113,294         2,-3        7,7,0,25
             15         48                          148,3,33,2
                                        (lead 1hx6w wet, strong air at K10)
K11:         ..                                     ,,32,3
K8->K12:     50.3       191,11          -1,0        4,12,0,3
K12:         48.9       180,0           1,-1        7,10,0,2
             49.6       152,334         1,-1.5      10,5,0,2
             49.4       171,352         0,0         8,10,0,2.5
             51.3       182.5,1.5       0           10,5,0,2
             28.4       157,338.5       0           10,3,0,1.5
K17: (flagged)          ..                          5,8,1.5,1
```

Figure 1. A CML listing of a recent survey. Ties are indicated by the tie operator (->) with the data for that tie appearing on the same line as its two stations. The data can continue down either leg of the tie. The station named on the line following a tie tells us which leg has been taken. Also, since the station number is the same in the tie, the stations being named on each data line must be the from-station. Note that station names can be abbreviated or omitted, as can the bearing in a vertical shot.

Data lines

Data are written as follows (see Figure 1): Each survey shot is on a separate line. A data line begins with a station name, followed by a colon, and then by the survey data. Survey data are in the order distance, bearing, and inclination. After this are the passage dimensions, separated by commas, in the order left, right, ceiling, and floor. Both bearing and inclination can consist of just a single number or a pair of numbers, corresponding to foresight and backsight, separated by a comma. If there is a backsight and no foresight, then a comma precedes the backsight. If the station name is clear from context, that is, if it is in numerical sequence with the preceding station, then it can be omitted. If the inclination is 90 then the bearing field can be omitted. The passage dimension fields can be omitted as well. The individual data fields can be separated by any number of spaces, tabs, or comments. Finally, departures from this basic format, such as adding a field for vertical distance, are indicated by special directives.

Directives

Commands (actually, they are called directives in the formal description of the language) are reserved words that begin with the pound sign (#). Directives serve many purposes and new ones can be added in order to give some new feature to the language or to the program. Example directives are #openFSB (followed by a number) which indicates that data from a particular Field Survey Book (FSB) follow, or #personnel (followed by some names) which is used to indicate the members of a survey party, or #close which directs the program to do a loop closure at a particular point in the data stream.

Comments

There are two ways of embedding comments in a line. A semicolon [;] means that the remainder of the line is a comment and is to be ignored by the computer. Additionally, any characters appearing between parentheses are also treated as comments. These parenthetical comments can appear almost anywhere in the data. The computer program considers them equivalent to a space, so they can go anywhere that a space can go, even embedded in a station name, but not within numbers or commands.

Station names

In the Mammoth Cave system, both CRF and the Central Kentucky Karst Coalition have adopted the convention of having three fields in their station names. First is the number of the Field Survey Book, followed by the one or more letters of the survey designation, and finally a station number. This is required since there are over 40,000 survey stations in the system and, therefore, no sensible means of creating unique survey designations using just letters. In addition, there are primed survey stations (A25') and primed surveys (A'25) and worse departures from good sense that have occurred over the years (we have Greek-letter surveys, a dollar sign survey, and several surveys that are just numbered). CML allows for most of this. When a full survey name is used, it is preceded by a dollar sign, such as $2344A10 or $1007AB'102. The Greek-letters, and other odd symbols, have to be spelled out, as $143Omega10. If it is clear from context what the proper FSB number or survey designation is, then those fields can be left out. For example, once station $2344A10 has been named on a line, then subsequent data lines can have just have the sequence A11, A12,... or 11,12,... or, since the stations numbers are in order, the station names on these subsequent lines can be omitted entirely. When present, the station name must be followed by a colon (:). (Think about it, without this rule, there would be no way for the program to determine whether the first number encountered on a line were a station number or a distance.)

Some station names do not conform to this format. For example, there is a set of high precision survey benchmarks through part of Mammoth Cave. These have USGS names, such as 'TT8W'. CML allows for this by permitting such odd station names to enclosed in single quotes.

Links and Ties

Connectivity can be specified in several ways. A survey shot that joins two surveys or that in any way connects two stations that are not in numerical sequence is called a tie. Ties have a special notation that allows both stations of the tie to appear on the same line as its shot data (i.e. A10 -> $804X1: ...). This makes for very readable, to the human, data files. However, ties can also be represented in the way common to most other survey data formats, by just naming the stations of the tie on successive data lines (Figure 2).

```
#openFSB 571
#location  Cutarounds in Lower Crouch
; R Brucker(*c)  W Rogers  R Eggers(b)
; 28 Nov 1970
; Compass Psu5-66156 Of Jack Hess
; Typed 2/17/73 By Wrc
#date                   701128
#compass                66156
#declination            1.4

#survey $571M1
$537L34:                22.2          S77.5E  -28      4,4,8,2
#no_elevation (no inclinations on the following shots)
$571M1:                 50            N28.5E           6,2,0,3
M2:                     17            N41E             1,7,0,4
                        30            N28W             11,5,0,4
                        50            N2.5W            7,4,0,3
M5:                     31            N2E              6,10,0,4
                        37            N3E              3,13,0,3
                        50            N15E             7,8,0,4
                        38            N6W              11,3,0,4
#elevation
                        25.7          N36W    13       6,7,0,3
;There is a sketch with L29 here.
M10:                    12.4          S53W    19       1,15,5,2
;                       ### Warning: NonInteger Station, removed by comment.
$537L27(.1):            ..                             0,0,0,0
;
#survey $571N1
$571M1:                 17            S25.5W  0        2,10,0,3
$571N1:                 28            S18E    0        6,7,0,2
N2:                     17.3          S11.5E  17       3,3,0,3
N3:                     ..                             2,4,1,2
```

Figure 2. CML also allows ties to be indicated implicitly by a change in the sequence of station names. For example, the shot between $571M1 and $537L34 is a tie. It has been found to be much easier to translate much of our older data into CML using this convention, as in this example. It should be clear by inspection that the station being named on each line is the from-station, as was the case in Figure 1. If there are dimensional data for the last station in a sequence, the two dots (..) signify that there are no shot data corresponding to that station (it is the to-station of the previous data line). The passage dimensional data can then follow the two dots. Note that bearings are given in quadrant notation, which is acceptable to CML.

It is also common in Mammoth Cave for two or more surveys to be joined by aliasing, that is by giving one or more names to the same station. These I call links and are noted in CML by the equal sign, (i.e. M34 = Y18 indicates that station 34 of an M survey is the same as station 18 of a Y survey).

CML allows forward referencing. That is, it is not necessary for a station to have been tied to the existing survey before it is referenced in the data. This is very useful as data can always be entered in the sequence used in the survey book. For instance, it is possible to enter a survey that begins in a dead end passage and continues out to a tie in that order, without the artifice of having to reverse the block of shots between the passage end (start of survey) and the first tie (although, see Figure 6).

Fixed Locations

The final aspect of a cave survey is fixing the cave in real world space. Any number of stations can be given a fixed location, again using the equal sign notation and specifying three-dimensional Cartesian coordinates. The Cartesian coordinates are enclosed in square brackets, (i.e. $12E1 = [6020,17001,486]). Assuming default units, this latter means that station E1 is 6,020 feet east and 17,001 feet north of some reference point and at an altitude of 486 feet. (The reference point for the Mammoth Cave data is a USGS benchmark outside the Carmichael Entrance.) This same notation can also be used to introduce leveling data into the data set (Figure 3).

```
; Entrance Passage
#feet
A1 = [,,0]          ; =datum (at entr.)
A3 = [,,2.4]
A6 = [,,3.3]
A8 = [,,1.8]
A10 = [,,-34.6]
A11 = [,,-52.3]
D1 = [,,-63.8]
D1 = "tomb"
C = "Collins Avenue"
C3 = [,,-35.1]
C4 = [,,-24.3]
C6 = [,,-4]
C8 = [,,-2.9]
C9 = [,,-2.5]
C12 = [,,-5.4]
C14 = [,,-7.2]
C16 = [,,-8.5]
```

Figure 3. Here is a short sequence of leveling data from Art and Peg Palmer's work in Floyd Collin's Crystal Cave. This is the same notation used for assigning fixed locations to stations except that the first two fields (easting and northing) are empty. It is also clearly noted, in a comment, that the elevations are not absolute altitudes. The elevation values would be processed as position estimates having some arbitrarily large weighting in the Z-space loop closure. The equal sign can also be used to associate a name to a station or survey.

A look inside

Implementation and language specification are two separate things. Were there no computer program to process CML data, the notion of a Cave Map Language would still be useful to us for purposes of record-keeping. Small programs or even commercial text editors could be used to convert CML data to formats readable by standard data reduction programs. However, the utility of a program that could parse the complete language is clear, and I set out to write one. Were caving not such a specialized thing, with relatively few participants, I might be so vain as to expect many implementations of CML developing over the years, just as there are many Pascal compilers. Instead, it is most probable that the program I have written to process CML data will remain, for a long time, the only one. Here is a brief explanation of its design and workings.

The program is written in very vanilla C. It is so transportable that development has been able to take place simultaneously on two machine architectures: Macintosh, using Apple's MPW C compiler, and on a PC clone, using Microsoft's Quick C and v. 5.1 optimizing compiler. There are occasional glitches. (The C language is not the ideal model of source code compatibility that people claim it to be. Implementors have taken liberties with C, such as adding new reserved words to handle DOS's segmented memory, that no one would ever dare to do with, say, venerable Fortran.) The interface is a command line one-- simple and old fashioned. On the Macintosh, it requires the MPW shell to run. There is presently no specialized data-entry program. Data entry must be with a text editor or some other means devised by the user.

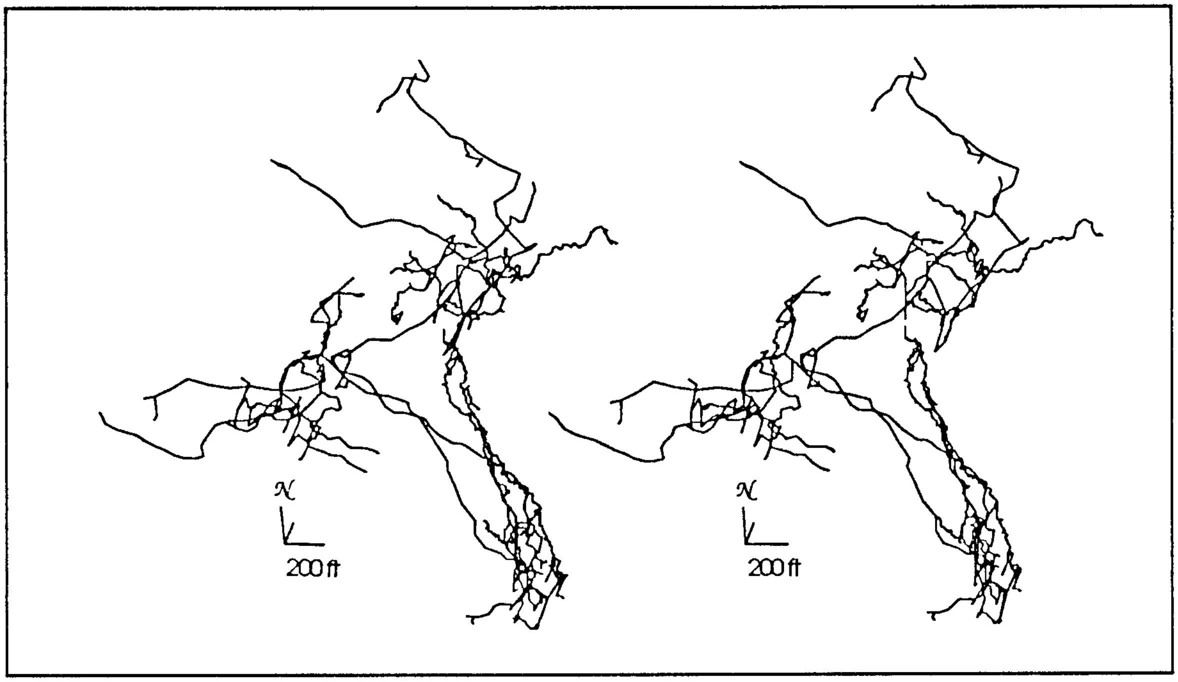

Figure 4. The data from Figure 2 appear roughly in the center of this stereo-pair plot of the complex area around the Unknown Entrance. It is intended to be viewed with a pocket stereo viewer. Vertical dimensions are exaggerated by a factor of 10. Cartesian data calculated by CML were imported into a commercial graphics program (Super3D by Silicon Beach, Inc.) in order to generate this figure. Any number of commercial applications are available for each of the major computer families. These provide an efficient way of generating high quality screen and hard copy graphics.

Two passes

The program is roughly modeled after an assembly language processor (an assembler). Like most assemblers, CML goes through the data twice. There is no magical requirement for this, a one-pass processor could just as well have been written. In the first pass, several key data structures are filled with information about the size and interconnectivity of the cave. Arrays of data structures describing each survey, link, fixed location, and tie in the data stream are made. By the end of the first pass, the memory requirements for further processing are known, as is the topology of the cave.

In the second pass, the survey data are converted to Cartesian coordinates and, if called for, loop closures are done. At its end, the various output files, such as graphics files for commercial applications (Figure 4), schematic diagrams (Figure 5), statistics, Cartesian coordinates, etc., are written to disk.

```
;------------------------- Field Survey Book $571 -------------------------

Survey $5710 (48)   Survey $571N (47)   Survey $571M (46)

01 <- $571M2        N1 <- $571M1*       M1 -> $571N1*
 |                   |*                 M1 <- $537L34
014 -> $4A0         N3                   |
 |                                      M2 -> $57101
015 -> $537L37                          M2 -> $825P1
 |*                                      |
015                                     M10 -> $537L27
                                         |*
                                        M11

;------------------------- Field Survey Book $624 -------------------------

Survey $624C (82)   Survey $624B (81)   Survey $624A (80)   Survey $624Z (79)


C1 <- $415V3        B1 <- $624Z2        A1 <- $624Z2*       Z1 <- $624Y15

 |                   |                   |*                  |

C17 -> $838C18      B7 -> $415X7        A8                  Z2 -> $624B1

                     |*                                     Z2 -> $624A1*

                    B8                                       |

                                                            Z6 -> $402T3

                                                             |*

                                                            Z7

Y1 <- $2#38
 |
Y15 -> $624Z1
 |*
Y27
```

Figure 5. We conceptualize Mammoth Cave as a network of interconnected surveys. The schematic diagrams generated by CML have the potential of supplanting the laboriously hand drawn ones that we presently use. CML has gathered all the ties of the M and N surveys of Figure 2, including ones that are noted in later survey books, and represented them in the form shown. The direction of the arrows indicates the direction of the survey shot in each tie. The asterisks flag dead-end strings, for the purposes of loop closure. The number in parentheses is the number of the survey that CML has assigned for internal use.

Virtual Memory

In five years, I am sure that virtual memory will be a service of every operating system. That is not the case now, however, and even basic memory manager functions, heap compaction and the like, are not available in unenhanced MS/DOS. The CML program, therefore, implements its own paged memory and memory manager scheme, using standard C library calls. The many data structures that are used in the course of a program run can be considered as arrays of structures but they are not really maintained in memory as simple contiguous arrays. Instead, they are grouped in blocks of either 64 or 2048 elements and indexed through a table that matches index with either pointer or location on disk. If there is adequate memory for a particular job, there may be no paging to disk. As more space for, for example, link records or Cartesian data is required, additional blocks are dynamically allocated in memory. Only if there is no room in memory, will an existing block of the same type (the least recently accessed one) be written to disk and its space in memory taken.

```
;Lover's Leap
#cff=-1
$956F5 = [-5266,7735,663]
$964A21:                158.5     12.2      0
A21':                   2         0         -90
A20:                    32.5      11        45.3
A19:                    8.9       60.5      35
A18:                    6.4       153.5     57
A17:                    10.8      142       0
A16:                    12        186       0
A15:                    10.8      191       0
A14:                    6.5       216       0
A13:                    3         306       0
A12:                    9.1       310       0
A11:                    15.3      270.25    0
A11':                   1         0         -90
A10:                    4.5       274       -78
A9:                     5.7       306.5     0
A8:                     13        270       0
A7:                     19.7      242       0
A6:                     21        219       0
A5:                     43.8      240.5     0
A4:                     24        265       0
A3:                     14.5      211       0
A2:                     14.5      203       0
A2':                    7.5       0         -90
A1:                     24.1      181       -17.8
```

Figure 6. Another set of existing data was originally encoded in a program that uses to-vector format. That is, the station being named on each data line is the station being shot to. The translation into CML is straight forward as this convention is also permitted in CML. Note that the last line in a survey sequence now contains survey data, instead of the empty data operator (..). If this string of data did not begin with a fixed location, then it would have been necessary to set off the first station with double dots, (i.e. A10 ..) to begin the to-vector sequence. The program that this data was originally prepared for could not handle forward referencing. This explains why the stations are in reverse numerical sequence. CML accepts this and considers only the shots F5->A21 and the shots involving the three primed stations as being out of sequence and, therefore, ties. Note, too, that the dimensional data are missing.

Internal Data Structure

Let me beg forbearance as I discuss some of the ways that data are organized internally in CML, usually a guaranteed yawner. Having a virtual memory system can promote downright licentious excesses. On the other hand, having total guarantee of sufficient space for any kind of storage, with little sacrifice in speed, allows a number of program efficiencies. For example, shot length is a parameter that is needed during loop closure and adjustment. Programs that are short on memory might be tempted to throw that value away after the polar-to-Cartesian conversion and recalculate it from the rectangular data when it is later needed. The CML program simply stores shot length along with the Cartesian data and so avoids a slew of floating point calculations.

Survey records contain fields for the FSB number, the letter designation of the survey, the number and range of stations, plus many other parameters. Survey records also store the array indexes that point to the link and station location data structures for that survey. Links, ties, and fixed locations share a common data structure called a link record. Every time a '->' or '=' verb is encountered in the data stream, a new link record is created and filled with information as to the from-station and, in the case of links and ties, the to-station.

Strings, a series of contiguous shots between two junctions, are a key concept in loop closure. The concept, it turns out, has wider utility. CML keeps track of strings by making a data structure for each string. In it are stored the beginning and ending station of the string, the total length of the string, and enough other information so that it is possible to trace the interconnectivity of the cave by manipulating these string data structures. Since there are far fewer strings than survey shots, the CML program gains considerable efficiency by adopting this strategy, instead of the alternative of tracing passages through the individual survey shots. The schematic diagrams (Figure 5) are made without having to consult the individual station data. The data structures describing surveys, strings, and ties (including links and fixed locations) are enough.

Conclusions

The value of a defined syntax-- a language-- for communicating ideas in computer science, and not just as a data processor, is also an old one. Algol and Pascal both began life as pencil on paper languages for communicating and were only later implemented in compilers. Admittedly, having to require a program as complex as CML has become to parse the data may prove to be an impediment to a wider utility of CML. However, in the small circle of Mammoth Cave Survey data, where it can be maintained, it will play, I suspect, its designated role.

Availability

The results of all of my work on CML, the language specification, the compiled images, and the source code, are in the Public Domain. They are, on the other hand, not yet readily accessible, for the simple reason that there are presently no reasonable means that would allow me to respond to requests for code or for support. Nonetheless, I will respond, to the best of my ability, to letters or to E-mail [MPARK@UTMEM1] and will endeavor for as open a distribution as I can arrange.

Acknowledgements

I can make few claims of originality in developing CML. Many of the ideas that I have implemented in CML have been discovered and used by others in their programs. Easy to read ASCII format, the notions of comments in the data, and directives are found in the widely used CMAP and in Cavelst, the CRF data reduction program of the 1970s, written by Will Crowther and enhanced by Bill Mann. Using free format data lines is a concession to the human but not a new one in programs for cave survey data. As I understand it, CMAP also started out with free format data lines only to later change to fixed length data fields.

I compliment and thank Tom Kaye, whose correspondence first taught me a lot about the algorithms of simultaneous closure. Equally, I thank Bob Thrun, who has supplied me with a wealth of information, code, and compiled programs, plus some deserving critiques.

TIPS AND TECHNIQUES

Blue Light in Caves
by Frank McNutt

After reading your tips and techniques from Vol. 7 #4, I thought that I'd make a small contribution. Regarding your point #4, blue light is supposedly more easily discerned than red by the human eye. That's one reason why some emergency agencies are switching their colors. There is more to this story, i.e. origin of this new technology, but I'm definitely out of space.

Refurbishing Suuntos
by George Veni

Problems with your Suunto that you can't fix? Have you taken it apart and done more harm than good? Don't throw that Suunto away! The Ben Meadows Company will repair and completely overhaul your Suunto (make it like new) for about $20 less than the cost of a new one. Basically they take out the internal case that contains the guts of the compass or clinometer (the site of most major problems) and replace it, meanwhile cleaning and fixing whatever else may need to be cleaned or fixed. In July 1990 the total cost of repair and shipping was $33. They accept VISA and MasterCard. Send your instruments to:

Ben Meadows Company
P.O. Box 80549
Chamblee, Georgia 30366
1-800-241-6401

They also sell Suunto compasses and clinometers with internal lights for $83.50 and $87.50 respectively.

How to Clean Your Suuntos
by Beth Webb

I am not a "super caver" or a "super surveyor" but here is a method for Suunto cleaning. Tom Kaye has put me up to writing this method up, for your information.

I am a super clean caver. This means I clean everything after a caving trip including tapes and instruments. Sealing the instruments before their first exposure to a cave is a high priority but eventually the seal is breached and you have dirt in the windows. Most surveyors ignore this problem until they are so dirty that reading them is like looking through dirty dishwater. They return them to the manufacturer with money and have them "refurbished".

I have found that with an air compressor you can prolong this "in-between" period. Take the offending instrument(s) and hold it under a stream of water (from the faucet). Really get water into them so that when you look through the windows, you see water bubbles along with the dirt. Take an air compressor (approximately 100 psi) and use the "air gun" attachment (or any one that has a controllable stream of air). Direct this stream of air in around the edge of the large plastic window of the instrument and depress this window slightly during the process. The dirt and water will be forced out (usually very quickly). Repeat the "air blowing process" 2 or 3 times at different spots around the window's seam and hang the instruments out to dry (with maybe a little sun) until all condensation disappears. Then re-seal the instrument.

This has worked on Tom's, mine, and Fred Grady's instruments. This does not completely solve the problem, but it definitely prolongs the trip and expense for the "refurbishment".

MAPS OF THE 1990 CARTOGRAPHIC SALON

report by George Dasher

JUDGES: Don Coons, Rutland, Illinois
Ann Strait, Calsbad, New Mexico
Bob Thrun, Adelphi, Maryland

ENTRANTS: There was a total of 47 maps entered this year. These maps were from: Guatemala, Mexico, Alabama, Alaska, California, Indiana, Kentucky, Missouri, New Mexico, Ohio, Oklahoma, Puerto Rico, and West Virginia.

AWARDS: 7 green - Honorable Mention
4 blue - Merit Award
1 Medal - Overall

CLASSES: short = 0 up to 500 meters or 1640 feet
medium = 500 meters up to 1.6 km or 1640 feet up to 1 mile
long = over 1.6 km and 1 mile
special = unique representations

MAPS:

Andy's Cave	Eddy County, New Mexico	John Brooks
Bat Cave	Carter Caves State Park, Carter County,	Horton Hobbs III
Bi-Level Cave	Woodward County, Oklahoma	Sue Bozeman
Bixby Landing Sea Caves	Monterey County, California	Peter Bosted
Buena Vista	Oaxaca, Mexico	Nancy Pistole GREEN (short)
Carey's Big Mud Cave	Anza-Borrego Desert Park, San Diego County, California	Bob Richards BLUE (short)
Catclaw Cave	Catclaw Cave Nature Preserve, Crawford County, Indiana	Keith Dunlap
Cathedral Cove Cave	Channel Islands National Park, East Anacapa Island, California	Bob Richards
Cenote Naharon	Tulum Q R, Mexico	James Coke BLUE (special)
Clancy's Cave	Meade County, Kentucky	James Greer
Colonnade Cave	Pendleton County, West Virginia	T. Ross and Charles Hoffman

Cottonwood Bat Cave	Blaine County, Oklahoma	Sue Bozeman
Cold Dunk Cave (La Cueva del Chapuzon)	Tala Jalisco, Mexico	John Pint
Cueva Mariposa	Opxaco, Mexico	Nancy Pistole
Cuevas de la Forja	Chupaderos Municipio de Zaragoza, Nuevo Leon, Mexico	Peter Sprouse
Davis Blowout Cave	Blance County, Texas	W. Ellicott and D. Pate
Doug Green Cave	Jackson County, Alabama	Pat Kambesis GREEN (medium)
El Captitan Cave	Prince of Wales Island, Alaska	Kevin Allred
E-Ticket	Arroyo Tapiado Canyon, Anza-Borrego Desert State Park, San Diego County, California	Nancy Pistole
Fieldhouse Cave	Pendleton County, West Virginia	National Youth Science Camp Expedition
Freudian Complex of the Chimney Lava Tube System	Siskiyou County, California	Lisa Wolff
Fuzzy Coon Cave	Menitee County, Kentucky	Horton Hobbs III
Grand Gulf	Oregon County, Missouri	Mike Sutton
Insanity Culvert Cave	Siskiyou County, California	Lisa Wolff
James Creek	Tarey County, Missouri	Jon Beard
Junction Cave	Valencia County, New Mexico	Mike Goar GREEN (medium)
Kicking Horse Glacier Cave	Garrison Glacier, Southeastern Alaska	Carlene Allred GREEN (short)
Las Cuevas de san Josecito	Municipio General Iqnacio Zaragoza, Nuevo Leon, Mexico	George Veni GREEN (short)
Lincoln Caverns	Lincoln County, New Mexico	Mike Goar GREEN (medium)
Lilburn Cave	Kings Canyon National Park, Tulare County, California	Peter Bosted GREEN (special)
Little Mud Cave	Arroyo Tapiado Canyon, Anza-Borrego Desert State Park, San Diego County, California	Scott Schmitz

Locust Creek Cave	Pocahontas County, West Virginia	Ron Simmons
Nai Tunich	Municipio de Poptun, Peten, Guatemala	George Veni BLUE (long)
Popcorn Pit	Tulare County, California	Bob Richards
Price Strike Cave	Wilburn Valley, Giles County, Virginia	Mike Futrell BLUE (medium)
Rehoboth Church Cave	Monroe County, West Virginia	Bob Frostick
Rio Camey Cave	Barrio Quebrada, Camuy, Puerto Rico	Russ Gurnee
Rocking Chair Cave	Eddy County, New Mexico	Dave Belski
Snowflower Pit	Jackson County, Alabama	Pat Kambesis MEDAL
Stairstep Dome Cave	Siskiyou County, California	Peter Bosted
Sunbeam Cave	Siskiyou County, California	Peter Bosted
Triple Eagle Pit	Charlie Martin Ranch, DeBaca County, New Mexico	Dave Belski
Widow Cave	Major County, Oklahoma	Sue Bozeman
Wishbone Caves I and II	Blaine County, Oklahoma	Sue Bozeman
Woolf's Lair Cave	Sierra County, New Mexico	Mike Goar
Zane Caverns	Logan County, Ohio	Horton Hobbs III

THE MINUTES OF THE 1990 MEETING
OF THE SURVEYING AND CARTOGRAPHY SECTION
OF THE NATIONAL SPELEOLOGICAL SOCIETY

The annual meeting of the Surveying and Cartography Section of the National Speleological Society was held on July 10th, 1990 in Classroom B of the Yreka Community Center, in conjunction with the 1990 NSS Convention, Yreka, California.

In attendance were: Rick Banning, Roger Bartholomew, Lee Blackburn, Rich Breisch, Barry Chute, Hubert Crowell, Don Conover, George Dasher, Bob Hoke, Pat Kambesis, Tom Kaye, Kirk MacGregor, Dale Pate, Nancy Pistole, Scott Schmitz, Bob Thrun, Carol Vesely, Harold Vodel, Jeanne Wittenburg.

Vice-Chairman Carol Vesely called the meeting to order at 12:33 PM. She asked for corrections and additions to the 1989 minutes, which had been printed in a 1989 Compass & Tape. There were none.

Treasurer Rich Breisch reported that the Section has $759.53, up $320 from one year ago. He stated that a lot of people are paying their dues for more than one year an that they are purchasing back issues of the C&T/ At present, the Section has 160 paid members, down a little - Rich believes - from one year ago.

Pat Kambesis asked how each Section member is notified when his or her membership expires. Each member's expiration date is listed on the mailing label. There was some discussion and many people expressed a wish that a more obvious technique could be used.

Tom Kaye gave his editor's report. The circulation of the C&T is currently just above 200 members. Tom stated that the C&T goes to eight foreign members and that these eight members cost the section more than all tho other members to mail. There was some discussion and it was generally agreed that the Section should absorb the costs of foreign members.

Tom further said that, within the last year or so, he has received many suggestions to print the medal-winning Cartographic Salon maps in the C&T. He wanted to make it very clear that - so far - he has received not so much as one of these maps and he requested that all winners send him copies of their winning maps, or parts of them.

Carol Vesely asked about the NSS' cave surveying book. George Dasher stated that this was a BOG project, not a SACS project. Nevertheless he badly needs SACS' help and was willing to give such a report. He generally said that he had completed yet another technical revision and that he now needs illustration, principally cave maps and pictures or drawings of people surveying. There will be a meeting concerning the cave surveying book Thursday at noon in the Cartographic Salon building at the Yreka Fairgrounds.

Elections were next. George Dasher stated that he had heard rumors that John Ganter wanted to be Vice Chairman next year, so that he could run the session during the New York Convention. No one else could make any comments. Barry Chute nominated Carol Vesely for Chairman. No one else was nominated and Carol was accepted by acclimation.

Rich Breisch nominated John Ganter for Vice Chairman. There were no other nominations and John was accepted by acclamation. Tom Kaye nominated George Dasher for Secretary. No one else was nominated and George was accepted by acclamation. Bob Hoke nominated Rich Breisch for Treasurer. There were no other nominations and Rich was accepted by acclamation.

Carol Vesely pleaded that everyone pay their dues. The meeting was adjourned at 12:54 PM.

Respectfully submitted by
George Dasher, Secretary

COMPASS & TAPE

Survey and Cartography Section
NATIONAL SPELEOLOGICAL SOCIETY

c/o 1732 Byron Street
Alexandria, VA 22303

Forwarding and Return Postage
Guaranteed

Address Correction Requested

N E W S
D A T E D
M A T E R I A L

If your name is circled
this is your last issue

NON - PROFIT ORG.
U.S. POSTAGE

P A I D

MERRIFIELD, VA
PERMIT NO. 993