A primary requirement of upper management is to be informed quickly and concisely. Executives at Virginia Electric and Power Company felt that this need could be partially satisfied by employing the recent color graphics technology. They requested a graphics system by which they could gain new perspectives from graphic presentation of data.

In response to this request, we developed the Management Information Facility (MIF). MIF is an interactive tool that provides a method for integrating several graphical techniques and information from many departments into a single display system that can provide decision-making information to upper management. SAS/GRAPH was chosen as the primary software tool to be used by each department for composition of specialized graphs because it allowed access to existing computer files as well as flexibility in presenting graphical data. In addition, the Interactive Chart Utility of IBM's Presentation Graphics Facility (PGF) was occasionally used for certain data types.

In this paper, we will outline the steps involved in the development of MIF and discuss its organization and use to the manager and the programmer.

Introduction

Upper management in any large organization has a need to receive critical information on costs and operation statistics in a timely and visual manner. Rather than reading a lengthy report or a table of statistics, the manager often needs the brevity and effectiveness of a graph. In a compact area, a graph can reveal a great deal of information. Through this line of reasoning, VEPCO executives envisioned a system of graphs of pertinent data that could be displayed in color on a computer terminal. The system needed to be designed so that no computer knowledge was needed and so that with the touch of a few buttons the required graph would appear.

In order to meet this broad request, a number of actions were set in motion. Management Information Services was asked to design the necessary system which became Management Information Facility. MIF provided the structure and means for formation, cataloging, and retrieval of graphs. Several departments were assigned tasks of assembling the data required and programming the graphs, and IBM 3279 color terminals and 3287 color printers were ordered for CRT display and hardcopy prints.

Color graphics in 1981 was a technological innovation. The field was examined, and graphics software were compared. We were interested in acquiring software that would allow relatively quick programming of graphs, yield professional-looking results, and require little training of personnel.

Selecting And Comparing Graphics Software

A number of people at VEPCO were already programming in SAS and were aware of the potential of SAS/GRAPH. However, there were also non-programmers to be considered. For these people, a package requiring no programming skill was essential. Therefore, to meet the needs of management and the needs of personnel, it was decided that the system would employ two software packages; SAS/GRAPH and IBM's Interactive Chart Utility. With both of these packages, graphs could be programmed and stored for future viewing.

SAS/GRAPH is an extension of SAS thereby offering all of the capabilities of a programming language. The data to be plotted can be derived from input statements reading existing files, from data modified in a dataset, or from a SAS procedure. Data need not be entered for the sole purpose of the graph. SAS/GRAPH also offers flexibility in the creation of the graph itself. A wide variety of fonts, the use of titles, footnotes, and notes, and the BORDER option give graphs a finished appearance as well as permitting necessary information to be added. Although SAS/GRAPH is not designed for extremely complex graphs but for simple plots and charts, we have found that by using move statements, rotated titles and footnotes, and the formatting of values, difficult graphs can be made.

In the business world, it is often necessary to view two perspectives simultaneously. The OVERLAY option of PROC GPLOT allows two or more plots to be superimposed. However, any resulting legend will be that of the last plot. So if a legend is needed, it is preferable to use the NOLEGEND option and substitute titles, footnotes, or notes to print the legend; otherwise, simply use the NOLEGEND option to blank out the default. In addition, the OVERLAY option is designed for graphs with the same X and Y axes. To create one graph with two plots and two Y axes, a number of manipulations must be made. The graph in Figure 1 is produced by the code in Figure 2. It utilizes draw statements to create symbols in a legend printed by title statements. The left Y scale is produced in the normal fashion; the right Y axis is printed with move commands in a title statement. Its vertical line is a result of HREF in the plot statement. Tick marks appear when data is plotted with plus marks against a month beyond December. And bold reference lines are created when near-equal data is plotted with points joined by a line.

Another such graph (Figure 3) involves creating two separate scales on one Y axis. Labels on the right identify the scales. In this case, on the lower scale is a bar chart, and on the upper scale, a line plot. This rotated graph allows two related graphs to be shown in conjunction. By using GPLOTT with the OVERLAY option, points are plotted for both the upper and lower scales. In the lower
Data can be read from any source. Like the not documented on a screen as is ICU. made extremely complex according to the skill when used by a non-programmer. With instructions and knowledge of the programmer. The SAS/GRAPH package, then, requires some programming and is not documented on a screen as is ICU.

IBM's Interactive Chart Utility (ICU) does not allow the very complex graphs that SAS/GRAPH does nor any in three dimensions. Its major value is that it requires no programming skill at all. The user needs to be aware of a hierarchy of menus. By selecting options from the needed screens, both the format and the data of the graph are created. A program function (PF) key allows the user to display the graph at any stage of its creation. The user can produce line graphs, surface charts, histogram bar charts, pie charts, or Venn diagrams by choosing the appropriate option.

We have found that ICU is especially useful when used by a non-programmer. With instructions and PF keys given on each screen and the availability of a tutorial, a non-programmer can fairly quickly create a graph. ICU is appropriate when a small amount of data can be directly input on the terminal. In fact, the utility is not capable of reading data from another source, definitely limiting its usefulness.

With this package, numbers are printed above bars in a graph without special manipulation. Also, as seen in Figure 4, two pie charts can be displayed on a single screen. A number of pattern statements and reference lines are available; however, only one font can be used. Titles can only be of a certain length, and the user has little control over the legend.

ICU lends itself to simple charts and plots from newly-entered data and allows the non-programmer access to computer graphics. SAS/GRAPH provides the flexibility of a language both in its graphs and its data manipulation. Data can be read from any source. Like the basic SAS package, graphs may be kept simple or made extremely complex according to the skill and knowledge of the programmer. The SAS/GRAPH package, then, requires some programming and is not documented on a screen as is ICU.

Concept and Development of MIF

With decisions made concerning software packages and their programming capabilities, a unifying structure was still required. We investigated various techniques to integrate these different graphic packages and to allow access to them in a format that was easy to understand and easy to use. The decision was made to make the access to the graphical displays through an interactive hierarchy of menu screens (panels). IBM's System Productivity Facility (SPF) with the new Dialog Management Services provided us with the basic software to accomplish this objective. SPF Dialog Management allows the user to display pre-defined screens (panels), maintain and display tables of user information, and interface with user-written programs or TSO Clists. SPF is designed to allow the user to move through a hierarchy of menu panels by selecting options which are displayed on the screen. In the development of Management Information Facility (MIF), we took advantage of this capability to provide structure for storage, description, cataloging, display, and maintenance of graphs.

Using MIF--The Manager

The manager who wishes to view any graph available through MIF follows a few simple steps to gain access to a list of graphs in the area of interest. From this list, he can select the particular graph to be displayed. The first step is to log on to TSO. Once she is in the READY mode of TSO, she keys 'MIF'. This will execute a TSO Clist: which invokes SPF and displays the first MIF panel (Figure 5). From this menu panel the manager may select a list of graphs from one of three major divisions of the company, may choose to exit from MIF and return to TSO, or may enter the MIF tutorial. This tutorial is another hierarchy of panels that contain information about the use of MIF. The manager selects the desired division by placing the number that precedes the division name in the SELECT OPTION field at the top of the panel. This action will cause a table of graphs for that division to be displayed (Figure 6). When there are more graphs available for this division than fit on the screen, the manager may scroll up by depressing the PF-7 key or scroll down by depressing the PF-8 key to locate the graph he wishes to display. Once she has found the description for the graph she wants to view, she moves the cursor to the SELECT CODE input area which precedes the DESCRIPTION and keys in an 'S' (select). The graph will then be displayed on the screen. After viewing the graph, the manager depresses the END key (PF-3) and returns to the table of graphic descriptions. He may select another graph or return to the division selection menu by depressing the END key once more.

At any point in this dialog the manager may enter the tutorial to receive information on how to use MIF by depressing the HELP key (PF-1). This will cause a panel from the hierarchy of tutorial panels to be displayed. Each panel in MIF is attached to a tutorial panel so that when the HELP key is depressed, tutorial information about the panel is displayed. From this panel the manager may move from one tutorial panel to another for a better understanding of any portion of MIF. An index of the tutorial is provided which may be reached by typing an 'I' in the NEXT SELECTION area at the top of any tutorial panel. To reach the top of the tutorial hierarchy she keys a 'T'. To return to the MIF panel from the tutorial, he depresses the END key.
How It Works

The list of graphical display descriptions that the manager views consists of two elements in an SPF table. These two elements are the description and a unique label (SYSTEM NAME). This label performs two functions for MIF. First, it is used as the key to the SPF table. And because it is unique, it can also be used by the programmer to link any description in the table to any manual documentation that is kept about that graph. At VEPCO, one group that is responsible for producing graphs uses the initials of the programmer and a sequential number for SYSTEM NAME (see Figure 6).

The two elements that are displayed on the selection panel are only two of some 19 fields contained in the SPF table. These other fields are parameters that are passed to a TSO Clist when the manager selects a particular graph for viewing. Based on these parameters, the TSO Clist performs one of three functions which correspond to the three ways that graphical displays are created.

The first way is to execute a pre-written SAS program from a dataset that is located on a 3350 disk drive. The name and location of this dataset are described by fields in the SPF table. The data for this program may be contained in the program or may be of a more dynamic nature and reside in another dataset. The information necessary for the system to find this dataset is also passed as parameters to the Clist from the SPF table. When the graph is displayed from a GOUT dataset using the SAS PROC GREPLAY, the necessary information to execute the program in TSO foreground is similarly contained in the SPF table. The third form of display is a graph originally created using the Interactive Chart Utility. This graph is stored in two datasets: one for the data and another for the format of the graph. To display these graphs the TSO Clist executes a COBOL program that makes calls to IBM's PCF to display the graph. The datasets necessary to display this type of graph are also described in the SPF table. The TSO Clist is quite simple. It executes the program (SAS, SAS GREPLAY, or PCF) to display the graph and may call upon SPF to display error panels when necessary.

MIF Hierarchy

Figure 7 illustrates the MIF hierarchy. Rounded boxes represent SPF panels and the rectangular boxes represent TSO Clists which may contain SPF commands to invoke SPF, display panels, update or display tables, or to exit from SPF.

The programmer who has created a graph to be viewed by management uses the MIF command to update an SPF table with the information necessary for MIF to display the graph. The graphs for each of the three company divisions represented in MIF are contained in three separate SPF tables. To use MIF the programmer proceeds in the same manner that the manager did to view a graph. When the programmer reaches the table of descriptions, there are four options in addition to the 'S' (select) that the manager uses to view the graph. To add a new graph to the table so that it may be viewed by management, the programmer places an 'I' (insert) in the SELECT CODE field that precedes the line where the description should appear. This action will cause a panel to be displayed (Panel 3.2 Figure 7.) which asks which type of graph is to be added (SAS, SAS GREPLAY or TCO). A TSO Clist displays a panel (5.1, 5.2, or 5.3) depending on the programmer's response. By keying into pre-defined and labeled input fields on the panel, the programmer supplies the parameters that the TSO Clist (3.1) will use to display the graph. To change the description or any other table parameter that pertains to an existing entry, the programmer places a 'U' (update) in the SELECT CODE field. A panel similar to the add panel will be displayed with the current parameters for the graph. Any of these parameters except the key field or the basic graph type can be updated. To delete a graph from the table, the programmer places a 'D' (delete) in the SELECT CODE field which precedes the description of the graph to be deleted. A CONFIRM DELETE panel (4.4) will be displayed to allow a second chance in case the wrong graph was being deleted or in case the user meant to select an option other than delete. To view all 19 fields in the table that pertain to a particular graph, the user enters an 'E' (expand) in the SELECT CODE field. A panel (3.5) will display and label all fields.

Advantages

The major advantage of MIF is ease of use. A manager can quickly learn how to display key information in a graphical format. The creator of graphs who has no JCL or TSO Clist experience can concentrate on developing graphs and not be concerned about creating methods for management to view them. Many graphs may be added without introducing the unnecessary complexity of trying to keep track of large numbers of items. As new divisions or departments begin to use interactive graphics, they may be added as an option on the first menu (Figure 5). These graphs will be stored in a separate table but will still be accessed through MIF. The capability for the creator of graphs to add, correct, and delete them from SPF tables keeps the manager up to date on what information is available.

Disadvantages

The main disadvantages encountered by the users of MIF is that the graph may take several minutes to be displayed once it has been selected. This time is unacceptable to many managers. If the graph could be created by the programmer and stored in some type of screen format rather than in a SAS database, then this objection could be partially overcome. A batch program to 'refresh' the graphs could be run in off hours and the managers could have fresh, quickly displayed graphs.

A second problem we have encountered is in the use of MIF for data that is not very dynamic. Data that changes monthly rather than
**FIGURE 4**
Side By Side Pie Charts
Created With I/OJ

![Pie Charts](image)

**FIGURE 5**
VEPCO MANAGEMENT INFORMATION FACILITY
SELECT OPTION ===>
1 POWER STATION ENGINEERING AND CONSTRUCTION INFORMATION
2 POWER OPERATIONS INFORMATION
3 COMMERCIAL OPERATIONS INFORMATION
‘ DISPLAY INFORMATION ABOUT MIF
X EXIT FROM MIF

PRESS PF-3 KEY TO TERMINATE MIF

**FIGURE 6**
MIF COMMAND INPUT ===>
SELECT CODE
DISPLAY DESCRIPTION
SYSTEM NAME
- CAPACITY FACTORS FOR SURRY 1
- CAPACITY FACTORS FOR SURRY 2
- CAPACITY FACTORS FOR CHESTERFIELD 5
- CAPACITY FACTORS FOR CHESTERFIELD 6
- CAPACITY FACTORS FOR MOUNT STORM 1
- CAPACITY FACTORS FOR MOUNT STORM 2
- CAPACITY FACTORS FOR MOUNT STORM 3
- CAPACITY FACTORS FOR NORTH ANNA 1
- CAPACITY FACTORS FOR NORTH ANNA 2

---

**FIGURE 7**
1.1 MIF COMMAND
   - INVOKE SPF
   - DISPLAY FIND OPTION MENU

1.1 DIVISION SELECTION
   (FIGURE 5)

2.1 DISPLAY TABLES
   - PROCESS SELECT CODE

2.2 DISPLAY TABLES
   (FIGURE 6)

2.3 TABLES OF DESCRIPTIONS

3.1 DISPLAY GRAPH

3.2 UPDATE SCREEN
   - INSERT GRAPH IN TABLE

3.3 UPDATE SCREEN
   - UPDATE GRAPH IN TABLE

3.4 DISPLAY CORRECTION DELETE PANEL

3.5 CLEAR DELETE

3.6 SHOW ALL 12 TABLE ENTRIES

4.1 GRAPH

4.2 DISPLAY GRAPH
   - INSERT SCREEN
   - UPDATE SCREEN

4.3 DISPLAY GRAPH
   - DELETE GRAPHS FROM TABLE

4.4 CONFIRM DELETE

5.1 INSERT OR UPDATE
   - INSERT GRAPH FROM SAS CODE

5.2 INSERT OR UPDATE
   - INSERT GRAPH "DISPLAYED" FROM 'OUT' DATABASE

5.3 INSERT OR UPDATE
   - INSERT GRAPH CREATED USING I/CV
on a daily basis may not be appropriate for a display system with several minute execution speeds. This data could be prepared and distributed as a hard copy to the appropriate management.

Securing Sensitive Data

Sensitive data may be protected several ways. One way is to protect the dataset where the SPF table is located. We protect these tables against update and allow update authorization to persons who are viewing the graphs. An alternate choice would be to protect them against reading and authorize access to managers who need to view the data.

We use a lower level protection so that certain graphs are protected and others are available to all users. The creator of the graph uses TSO to enter and maintain the source code. The source code, data, and ICU formats are kept in datasets which follow the TSO dataset-naming convention. This means that the first node of the dataset name is the TSO USERID of the creator. The TSO PROTECT command is then used to secure these datasets. We have written an additional TSO application named TSOSEC for this purpose. TSOSEC presents the user with a hierarchy of menus. When the user has responded to the options and questions on the panels, TSO executes the TSO PROTECT command. The user need not know how the command works or even the command syntax. He/she need only respond to questions such as the dataset name, who will be authorized access, and what type of access will be allowed.

Printing MIF Displays

Our SAS graphs are not prepared using the SAS3279 device driver. We use a modification to the universal device driver so that we can use PFG to write our SAS graphs to the IBM3279 color terminals. This allows us to add a line of text to the bottom of the display screen that enables the manager to enter printer ID, copy size, and the number of copies to print. If the manager chooses, he/she may enter the name of IBM 3287 color terminal in this text line and the display (minus the text line) will be transferred to the printer. An ICU graph may be transferred in a similar fashion using a PF-key. It is hoped that the SAS Institute will provide this feature on a future release of SAS GRAPH.

Conclusion

Management Information Facility provides a key service to managers, programmers, and non-programmers. Without MIF, each department would house its own isolated set of graphs. MIF offers the computerized communication needed in a large company. It consolidates graphs from many departments, yet allows departmental separations and restrictions of access to data. Whether graphs were created with SAS code, PROC GREPLAY, or ICU is immaterial. All elements necessary for display on the color terminal are stored in the facility and with the keying of one letter are inserted into a TSO Clist.

The facility frees managers of the need for technical computer knowledge and provides them with color graphs on pertinent subjects with little time lag. They do not have to request someone to write a program to answer their questions or for someone to bring the graph to the screen. Therefore, MIF offers upper management the opportunity for quick graphic displays of data to aid in their decision making.

Interactive Chart Utility is available as a facility for IBM's Graphic Display Data Manager and Presentation Graphics Feature.