

Your Map to the Sea of GUIs: A Primer on Moving SAS® Applications Across Graphical Operating Systems

James Hefner, Entergy Corporation, Beaumont, TX

ABSTRACT

Currently, most SAS Software application developers have only one or two graphical operating systems (such as Microsoft® Windows™, or OSF/Motif™) to support. However, the pending release of the SAS System® for the Apple® Macintosh®, and the introduction of new hardware and software such as the PowerPC, PowerMac, and Wabi, means that application developers may have to support two or more graphical operating systems.

This paper is intended to assist application developers, both in the teaching of the fundamentals of graphical operating systems, and in the moving of SAS/AF® and SAS/EIS® applications from one operating system to another.

INTRODUCTION

If you are a SAS application developer, you may currently be working on one, or maybe even two, platforms and operating systems. You may be programming for IBM® Mainframe, or Microsoft Windows, OS/2®, UNIX®, or DOS, or a combination of the above. However, as you will see shortly, recent developments could cause you to have to "open up" your areas of application development. This means that you will have to become familiar with more than one or two operating systems. The easiest way to do this is to work through a graphical user interface (GUI) on every platform and operating system you support. Then, it will mostly be necessary to deal with the differences in file structure, and display and output devices. Consider the following developments:

PowerPC and PowerMac

These new RISC-based machines are jointly developed by Motorola® Inc., IBM, and Apple. They are slated to offer performance that is as good as, or better than, the current Intel lineup. The PowerPC's PowerOpen operating system should be able to run Windows, Windows NT, OS/2, Macintosh, and UNIX applications unmodified (using SoftPC to run Windows & OS/2 apps). Current plans are to offer these new machines at prices that are highly competitive with the current top-of-the-line offerings by IBM PC manufacturers and Apple, not to mention UNIX workstations. This could mean a change in the platform you are currently using, as well as the ability (or need) to be able to use and write applications using any of the five operating systems.

The SAS System for Macintosh

With the introduction of the Quadra and PowerMac, and the SAS System's full support for a graphical user interface (GUI), it has at last become feasible to offer a SAS System for the Macintosh that will be accepted by Mac users. Since many Macintosh computers in the corporate and educational fields are networked with other computers, it may become desirable or necessary to develop your SAS applications for the Macintosh platform.

Operating Systems for Multiple Platforms

SunSoft®'s Solaris®, Novell®'s Unixware® and NeXT®'s NeXTSTEP® 32-bit UNIX operating systems are now available for Intel® 486 and Pentium™ computers. Microsoft Windows NT™ for Intel 486 and Pentium computers is also available, with future releases planned for MIPS, DEC® Alpha, and other RISC

machines as well. A pair of new products, called Wabi (Windows Application Binary Interface) and SoftPC, will allow you to run Microsoft Windows 3.x applications unmodified on the PowerMac and Solaris, NeXTSTEP and other UNIX platforms. These developments also mean that you and your users may be running more than one platform or operating system. This calls for greater care in designing user interfaces to be more general in layout, and attractive enough in appearance to be satisfactory for users viewing the best applications commercially available under Windows, Macintosh, Motif, and NeXTSTEP.

New Graphical Operating Systems

In addition to the platforms mentioned above, Apple and IBM are currently working on the Taligent operating system, which will have an object-oriented, graphical front end. IBM is also discussing porting its object-oriented OS/2 2.x Workplace Shell to a new version of PC DOS® and AIX®, IBM's version of UNIX (to be called Workplace OS). Apple is also talking about porting the Macintosh interface to DOS and UNIX platforms.

All of these new platforms and operating systems offer strong support for networking. This will enable them to access and share SAS datasets and applications with possible existing platforms, as well as one another.

Why should I care?

As an application developer in a large corporation or learning institution, you may not feel it necessary to be good at designing GUIs. You may also feel that it is not necessary to write applications that are consistent with the current interface guidelines, since your users will "take whatever comes." If you do feel this way, consider the following:

- All graphical user operating systems, and their applications, are striving to present a consistent user interface. If your applications are not consistent, your users may perceive them to be obsolete, or at the very least, difficult to use.
- All of these operating systems are multitasking, allowing the user to toggle between an application and other programs. A poorly written application will "stick out like a sore thumb."

It is the intention of this paper to demonstrate that it is not difficult to write applications that effectively use a graphical user interface. In addition, once you learn one interface, and are familiar with the underlying principles of GUI's, it is not difficult to make the transition to another interface. Developing an interface for use across multiple platforms requires a knowledge of GUI conventions. A good tool for multiplatform application development, such as the SAS System, is required as well.

GRAPHICAL USER INTERFACE PROGRAMMING

If you have already written SAS/AF applications in a previous version of SAS, converting to a graphical user interface can be as simple as writing PMENUs to create menus and dialog box buttons, and replacing check fields and text with either radio knobs or check boxes. Also, context-sensitive help screens for each application should be written, using SAS/AF CBT entries. Valid entries should be entered as the LIST field attribute for each text and number entry field. For field entries that are not self-explanatory, such as FIPS codes, write a SAS/AF HELP entry that lists all of the valid entries, and their corresponding codes. For example, a help screen can be written listing all of the FIPS

codes and their corresponding states in a HELP entry titled STATES.HELP. The following line is entered as the List field attribute for that field:

```
\ ? = STATES.HELP \
```

This will call up the help screen whenever a "?" is placed in that field, and the user presses Enter.

You can also take a more radical approach, and replace the PROGRAM and MENU entries with FRAME entries. Then, take full advantage of all of the graphical widgets we will be seeing shortly. Another alternative is to totally rebuild an application using SAS/EIS.

Whatever approach you decide to take, bear a few things in mind. Character-based programs are usually linear in flow. This makes debugging and flow-charting them fairly easy. GUI programs, on the other hand, are event-driven; many, many variations on program flow are possible, based on the user selection and configuration. Expect to take more time debugging a GUI program, and be aware that changing your current testing program to a graphically oriented one may be necessary. Even simple things such as screen colors and fonts can vary widely, depending on the system settings. The SAS System's default settings on each platform are intended to make porting an easy task. However, your application may be taking advantage of features such as hardware or operating system fonts and icons, that do not port well. Also, your users may modify the default SAS System settings.

Interface design also plays a much greater role in GUI applications. Character-based programmers are "writing scripts" for plays, while GUI programmers are, in essence, "set designers". GUI application users are given a greater variety of choices, and the appearance of applications becomes as important as the content.

In order to write SAS applications that are consistent with applications currently being marketed, a knowledge of current GUI interface standards is needed. Users are less tolerant of inconsistent interfaces as commercial applications become more polished and uniform in their design. Currently, two interface standards dominate the graphical user interface scene: Common User Access™, or CUA™, endorsed by IBM and used by applications written for OS/2, OSF/Motif, Windows, DOS, and IBM Mainframe; and the Macintosh interface. (Since COSE, a group seeking to unify the various UNIX varieties, has chosen OSF/Motif as the basis for its interface standard, expect OPEN LOOK to begin to fade in popularity. Although an attractive interface, OPEN LOOK is neither CUA nor Macintosh compliant in its appearance or user interaction.) Publications that give guidelines on interface standards are listed at the back of this paper.

Because of the SAS System's MultiVendor Architecture™, SAS Institute is able to port 90% of the SAS System from one platform to another without modification. In moving your SAS/AF and EIS applications from one platform to another, you will also find that 10% (worst-case, probably much less) of your application code needs to be changed. The actual amount of change needed depends on how much you access files, use display drivers, and send output to printers. Another important factor is how much you do inside SAS Software, as opposed to calling outside applications and system services.

FEATURES GUI'S HAVE IN COMMON

There are some basic elements, or features, that make up all graphical user interfaces. Although their appearance and the user

interaction with them may vary somewhat, their basic function is the same.

The SAS System is a good tool for application development on multiple platforms because SAS automatically takes advantage of the features of each interface. Your main job is to ensure that the window and menu content, and the behavior of your applications, is consistent with other software on that platform.

Note: In order for this approach to work, all of the platforms you intend to support should have SAS version 6.07 or greater. This requires either upgrading SAS 6.04 for DOS users to Microsoft Windows and SAS Software version 6.08 for Windows, or giving them remote access to the SAS System running on another platform through a terminal emulator that provides support for window graphics and a mouse. Also, all application development should be done on the lowest version of SAS software requiring support— an application cannot be developed using SAS software version 6.08 or 6.09, and moved to SAS software version 6.07 on another platform!

Throughout this paper, five GUI's will be compared: OSF/Motif 1.2, Microsoft Windows 3.1 & Windows NT, OS/2 2.x, Macintosh, and GEOS™ for the Commodore® 64-128. SAS software now runs on all but the last (and it will demonstrate how these features are common, even to an 8-bit operating system.)

Desktop



The desktop is comprised of three elements: the Background, Windows, and Cursor. Every graphical user interface starts off with these three elements. The background is the area on which windows and icons are overlaid. On most graphical operating systems, the color of the background can be customized. A pattern of dots, or even a bitmapped picture or "wallpaper" can also be applied to the background.

Notice how each of the windows shown here has a Title Bar with the window name across the top. Notice also the Size Handles in the corners of the windows under Motif and Windows. (OS/2 also has size handles, although they are not visible. Macintosh uses a zoom box for resizing windows, which will be presented later.)

A special type of window that all GUI's provide is a Dialog Box. Dialog boxes are used to prompt the user for input. Dialog boxes are fixed in size, and those without title bars are fixed in position. They pop up over the current windows, and provide the user with options or dialog text and an OK, Cancel, or other push buttons. SAS/AF screens can be set to appear as dialog boxes by setting the Window Type to Dialog or Dialog Box on the general attributes screen, and assigning a PMENU to it. The GUI's shown here also have standard, or common dialog boxes that are used by most applications, as will be shown later.

In OSF/Motif, it is the X-Window hardware and software that provides the background, basic windows (without the size handles in the corners), and cursor. With the other four GUI's, it is the operating system itself that provides the background, basic windows, and cursor.

Widgets

Once the background, windows, and cursor are displayed, the rest of the interface is composed of images (either bitmapped or vector), fonts, and widgets. Of these three, widgets vary the most between interfaces.

Widgets are objects composed of a bitmap display and code. When the user interacts with a widget with either the mouse or keyboard, the code in the background determines its behavior. SAS/AF FRAME entries and EIS applications are composed of widgets (both operating system specific and SAS), and text. The behavior and appearance of each widget are determined by a combination of operating system settings, SAS software settings, and characteristics given by the SAS application developer.

OSF/Motif	Windows	OS/2	Macintosh	GEOS
				
Print	Print Manager	Printer	Image Writer	Comm. Control

Icons (above) are used to represent an object, such as a file or window; an action, such as printing or deleting; or an attribute, such as bold or underline. Icons are sometimes shown on the face of a push button, in which case the user simply clicks on the icon to select it. (The icons on SAS/ASSIST® screens are one example.) Otherwise, it is seen on the face of the window or background. In this case, the user clicks on it once to select it, and double-clicks on it to start or run the object. (This kind of behavior is used most often for running or restoring an application or window. Some icons that appear as shown are single-clicked in order to select and execute them. When this occurs, they are being used like hypertext, to jump from one area or window to another. The difference in content should be apparent, but sometimes isn't. In writing applications using SAS software, this difference is well maintained.)

Along with its built-in icons, on some platforms the SAS System also allows you to add icons of your own. If you decide to do so, this may make it more difficult to port your application. SAS Software icons take on the size and color specifications of the current hardware and operating system.

OSF/Motif	Windows	OS/2	Macintosh	GEOS
				(None)
				(None)

Maximize Buttons (top, above) appear at the right end of the title bar in CUA applications. Clicking on them enlarges an application window so that it fills the entire screen or desktop. Once a window has been made full screen, the Maximize Button is replaced by a **Restore Button** (bottom, above), which can be clicked on to restore it back to its previous size.

Macintosh windows do not have Maximize and Restore Buttons. However, the **Zoom Box** shown can be clicked on once to maximize a window, then clicked on again to restore it back to its previous size. Zoom Boxes appear at the right end of the title bar. In addition, the **Size Box** () can be pressed, held, moved, and released to resize a window. Size Boxes appear in the lower right hand corner of the window.

OSF/Motif	Windows	OS/2	Macintosh	GEOS
			(None)	(None)

Minimize Buttons (above) appear at the right end of the title bar, next to the Maximize/Restore Button (if there is one) in CUA applications. Clicking on them once shrinks a window down to an icon; double-clicking on the icon restores the window.

OSF/Motif	Windows	OS/2	Macintosh	GEOS
		(Appearance Varies)	(None)	(None)

Clicking once on the Control-menu Box (above), causes a pull-down menu to appear. This menu has commands for resizing, moving, maximizing, minimizing, and closing windows, as well as

switching between applications, and is useful for keyboard users. Control-menu boxes appear at the left end of the title bar.

Some third-party applications (such as Norton Desktop™ for Windows) allow you to add Desk Accessories or Applets to the Windows Control Menu. Then, the Control Menu also serves the same purpose as the Apple Menu on the Macintosh, and the geos menu on GEOS.

OSF/Motif	Windows	OS/2	Macintosh	GEOS
(Double-Click Control Menu Boxes)				
				

Double-clicking Control Menu Boxes on CUA applications, or clicking on the **Close Box** (above) on Macintosh and GEOS applications causes the window to be closed. Close Boxes appear on the title bar of Macintosh and GEOS windows.

OSF/Motif	Windows	OS/2	Macintosh	GEOS
 Choice 1	 Choice 1	 Choice 1	 Choice 1	 Choice 1
 Choice 2	 Choice 2	 Choice 2	 Choice 2	 Choice 2

Radio Knobs (above) let you make one of two or more choices. You can only make one selection; clicking on Choice 1 above toggles the Choice 1 Radio Knob on, and the Choice 2 Radio Knob off. One must be selected; you cannot choose none at all.

In setting up choice groups on SAS/AF program screens, if more than one pair of fields is assigned to a choice group and the action field list attribute is **blank**, radio knobs are automatically used. If radio knobs are to be placed on a FRAME entry screen, the SAS/AF software will not allow fewer than two radio knobs to be used. Therefore, SAS/AF helps to insure that radio knobs are used correctly.

OSF/Motif	Windows	OS/2	Macintosh	GEOS
 Not Checked	 Not Checked	 Not Checked	 Not Checked	 Not Checked
 Checked	 Checked	 Checked	 Checked	 Checked

Check Boxes (above) allow the user to choose desirable options. There may be one or more check boxes in a window, and their behaviors are totally independent of each other.

In setting up choice groups on SAS/AF program screens, if only one pair of fields is assigned to a choice group and the action field list attribute is **blank**, a check box is automatically used. FRAME entries also allow placing one or more check boxes in each FRAME window, with each being independent of the other. SAS/AF helps to insure that check boxes are also used correctly.

As a SAS application developer, YOU must be careful to insure that radio knobs and check boxes are used as they should be. To ensure that the user selects one of several choices (of printer destinations, for example), use radio knobs. To allow users to check off one or more options that are available, use check boxes. Their use is usually NOT interchangeable.

OSF/Motif	Windows	OS/2	Macintosh	GEOS
				
				

Push Buttons (above) appear on dialog boxes and windows. Push Buttons allow the user to take immediate action—OK'ing or canceling a dialog box, for example, or pulling up another window. The top button in each case is the highlighted or default button.

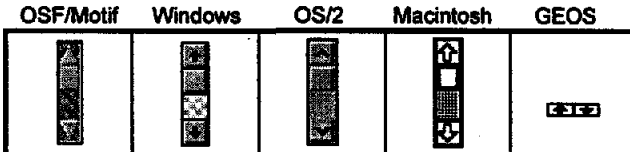
Notice how the default button has a black border around it on the Windows, OS/2, and Macintosh Interface, and a depression around in on Motif interface. (Although the OK button under GEOS is not highlighted, it acts as the default button when the user presses Enter.) Since the OK button is usually the first

button on the left, (and is also usually the default button,) this action is consistent across all of these operating systems. In addition, under all systems, you can move the default selection (or focus) around on the window by pressing the Tab key.

If you assign a PMENU to a SAS/AF entry, and give it a Window Type of dialog or dialog box in the General Attributes window, this will cause your menu to appear as a series of push buttons at the bottom of the dialog box.

A push button can also be added to a PROGRAM entry window. Create a field on the window, and give it an Alias and a Field Type Attribute of either PUSHBTNC for fields that return character values, or PUSHBTNN for fields that return number values. Enter the text you want to appear on the face of the button in the Initial Field. If you also want an icon to appear on the face of the button, enter the icon number in the List Field Attribute, with a "\ " on each side. For example, for icon number 4 to appear on the push button face, you would enter "\4\" as the List type attribute for that field. For FRAME entries, you simply place a button or icon on the window, and give it an alias.

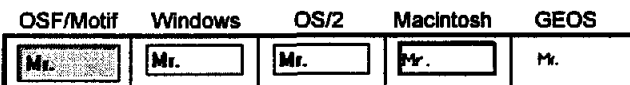
In the source window, you enter the SCL code you want to execute when the button is pressed, with the Alias of the field as the label before it, and a RETURN; statement after it.



Scroll Bars (above) are used to scroll around in windows, lists, or ranges of data values. They can either be vertical, as shown, or horizontal. Scroll arrows appear at each end of the scroll bar, and allow the user to scroll one line at a time. (On some graphical user interfaces, such as NeXTSTEP and the Commodore Amiga® Interface, the scroll arrows are placed together at the bottom of the scroll bar.) You can also grab the thumb button, or slider, and move it up and down to a particular spot on the scroll bar. Also, you can click on the face of the scroll bar itself to move up or down one screen at a time. (Note that GEOS has no scroll bar objects, only scroll arrows.)

A Slider (not shown) is a variation of a scroll bar, found in FRAME entries. Visually, it has a different looking thumb button, or slider, and no scroll arrows. It allows the user to input numeric values within a specified range, with a specified interval, by sliding the thumb button to a particular point on the slider.

Starting with SAS software version 6.06, scroll bars are optional on the edges of SAS/AF windows. You can also have scroll bars appear on the edges of all SAS/FSP® windows. Starting with version 6.08, scroll bars can be placed on FRAME entry windows, and be used to scroll through the contents of another widget.



Text Entry Boxes (above) allows the user to enter & edit a string of text. If the user is to view, and not edit text, then use a Label.

List Boxes (not shown) are multiline text boxes with a scroll bar on the right side. They allow users to scroll through a list of provided values, and select the value desired. Users are not allowed to edit current values, or enter new values.

Combo Boxes, (not shown) as the name implies, are a combination of text entry boxes and list boxes. Users may type in

a value, or click on an arrow button (or control) next to the text entry box to pull down a list of values, and select from the list.

Extended Tables and Spin Controls are other variations on the list box.

Notice how GEOS does not display a text entry box. SAS windows built with PROGRAM entries are the same way— in order for text entry boxes, list boxes, and Extended tables, to appear as shown, use FRAME entries to create applications.

Fonts

Fonts are an essential part of graphical user interfaces, and are divided into two types: soft fonts, and scalable fonts. Fonts can also be either proportionally spaced (with equal area between letters, like the text of this paragraph), or fixed-spaced (equal distance between characters, as on a character-based display).

Soft, (or bitmapped, fixed size) fonts are most often used for generating displays. These fonts can be generated quickly and are hand-tuned for best display quality. However, they take up a lot of disk space, and cannot take advantage of many of today's high resolution output devices. They also offer a fixed number of font sizes.

Scalable (or vector) fonts allow for an infinite selection of font sizes. When printed, scalable fonts will print with the same resolution or quality as the output device. PostScript, Bitstream and True-Type are the most common scalable fonts, with PostScript printing being available on all the operating systems mentioned here, and Display Postscript being used by NeXTSTEP and other UNIX platforms.

SAS Institute provides a fixed space display font for generating SAS/AF and other windows, and its own scalable fonts for use with FRAME entries and SAS/GRAPH®. The fixed-space display font can be replaced with a fixed-space system font, and the operating system's resident scaled fonts can be used for FRAME entries and SAS/GRAPH. If you do so, your applications will not be as easily moved across platforms.

Menus

Menus are another important feature of graphical user interfaces. Decisions on how the menus are to be arranged will play an important role in how users perceive your applications. A well designed, consistent menu will make your application easy to use. A poorly designed menu will have your users digging through menu pull-downs, trying to find the command they need.

In arranging your menus, it is important to remember not to make any item more than three levels (or clicks) deep. Otherwise, your menu will be considered too slow and difficult to navigate. Also, all items in the main menu should show a pull-down menu. Items that require immediate action should use a push button instead. The SAS system menus, along with other commercial and shareware software, are a source of ideas for menu layouts.

OS/2, OSF/Motif, and Windows menus all use mnemonics; and all of the interfaces mentioned here use accelerate keys. Mnemonics are the underlined characters found in a menu or dialog box, and allow the user to select an item by pressing the Alt key and the underlined character at the same time. Accelerate keys are keys you press to immediately execute a menu item. Using F1 for Help, and the editing keys mentioned later, are examples of accelerate keys. Appendix A lists a sample PMENU that includes all of these features, and other feature and layout ideas. Another benefit of using the SAS system for menu creation is that features which are not supported by the current operating system are simply ignored, without any errors.

Cut-and-Paste, DDE, OLE

All of the operating systems shown here give you the ability to cut or copy text or graphics from one area or application and paste it to another. The data resides on a clipboard and, on some operating systems, can be stored in an album or scrapbook for future pasting.

Microsoft Windows and OS/2 have the ability to create a link between two applications, where the data in the linked application is changed every time the original data is changed. This is called Dynamic Data Exchange or DDE, and requires that both applications be active in order for the updates to take place.

Another alternative is to include data in another application as an object. (An object in this case is data, along with enough code so that it is aware of where it came from, and how to display itself.) Then, in order to update the object, the user simply double-clicks on it to call up the application that created it. This is called Object Linking and Embedding, or OLE.

Macintosh System 7 also offers a similar technology called OpenDoc. Since the SAS System supports DDE and OLE under Windows and OS/2, it will probably also offer support for Publish and Subscribe, OpenDoc or OLE on the Macintosh.

Similar technology for sharing data is also available on UNIX platforms. An example is the Object Management Group (OMG), and the Inter-Client Communications Conventions (ICCC). The support is not compatible across platforms, however.

Managers

As the name implies, managers are used to regulate, or manage, constantly changing output. It is becoming familiar with the location of, and learning the use of widgets, managers, control panels and tools that is the first hurdle most users have to overcome in moving to a graphical user interface. The location of managers, control panels and tools has not been standardized, and varies widely.

For example, the SAS System provides a Graph Manager Window for working with your SAS/GRAPH output, and an Output Manager Window for managing text output. Also, Microsoft Windows provides a Program Manager for displaying and editing a collection of application icons, a File Manager for managing disk files, and a Print Manager for managing printer queues.

Control Panels or Preferences

Control Panels are used to make occasional changes in settings. These settings include adding or removing fonts, changing the desktop background, color scheme, date and time, adding and removing hardware drivers, and other system settings. Because control panels allow your users to make major system changes, they give users the ability to make applications difficult to debug and create errors.

The distinction between managers, control panels, and tools can be somewhat blurry, and the names of programs that do the same thing vary across operating systems.

Tools, Desk Accessories, Applets

All graphical operating systems are equipped with a collection of tools, desk accessories, and applets (scaled-down applications).

Tools are small programs, used to perform necessary system tasks. These include managing network connections, performing hard drive backups, and repairing and maintaining hard drives and floppy disks.

Desk accessories, or applets are provided for the users benefit and enjoyment. All of the following are available for each of the platforms mentioned here. Those not included with the operating system are available as freeware, shareware, and commercial products:

- | | | |
|-----------------------------|------------------------------|----------------------------|
| • Clocks and Alarm Clocks | • Calculators, various types | • Notepad and Text Editors |
| • Calendars | • Icon Editors | • Games/Demos |
| • Screen Savers | • Word Processors | • Message/Mail Services |
| • Paint Brush Programs | • Terminal Programs | • Clipboard Viewer/Manager |
| • Character/Keyboard Mapper | | • Databases |

Some applets that are bundled with the operating system are intended for the casual user only.

Drag-and-Drop

Drag-and-Drop gives the user the ability to perform an action by clicking on an icon, and dragging it onto a window or another icon. For example, a user can print a file by clicking on its icon in a directory window, and dragging it onto an icon of a printer. The user can also delete a file by dragging its icon onto an icon of a trash can or shredder. OS/2, with its object-oriented user interface (OOUI), makes extensive use of drag-and-drop.

SAS Software for Windows and OS/2 can be executed by dragging the icon for the SAS file you wish to run onto the SAS System icon. OSF/Motif supports drag-and-drop of text from one application to another, as well as icons.

GUI Shells

To most users, the Program Manager that appears when you first start Microsoft Windows is Windows. However, Program Manager is just another Windows application, being used as a GUI Shell. Program Manager can be replaced as a GUI shell by File Manager, MS-DOS Executive, Norton Desktop, or any of the other various Program Manager replacements that are available. OS/2's Workplace Shell, Macintosh's Finder, and the GEOS Desktop on GEOS are also GUI shells that can be replaced by alternative GUI Shells. (Under COSE, Motif will use a shell called Dashboard. It is similar in layout and concept to what Hewlett Packard® offers for its UNIX platforms and as a Program Manager replacement with the same name under Windows.)

Common Keystrokes, Mouse Movements

Applications on these operating systems share other keystrokes and mouse movements besides the ones mentioned earlier. For example, double-clicking on a word in a text editor or word processor highlights the entire word, allowing cutting or copying.

Looking at the lower left-hand corner of an IBM PC, Macintosh, and C=64 keyboard, there is a  key, a  key, and a  key, respectively. Realizing that these are Command Keys that you press along with another key to issue a command, the following editing commands become the same for all these GUIs: (Command Key)Z Undo (Command Key)N Normal text (Command Key)X Cut (Command Key)I Italic text (Command Key)C Copy (Command Key)B Bold text (Command Key)V Paste (Command Key)U Underline text. In addition, holding the Command Key down while clicking with the mouse allows you to select more than one item at a time.

Common Dialog Boxes

As mentioned previously, graphical user interfaces have standard, or common dialog boxes that are used by most applications, including SAS Software. These dialog boxes are used for opening & saving files, printing, and system messages. Some operating systems also offer additional standard dialog boxes for setting object attributes and finding text strings. Common dialog boxes make it easier for the developer to write a new application. They also make applications easier to use, since all of these actions are shared among applications, and are made consistent by having consistent dialogs presented each time. These dialog boxes are modal, meaning the user must respond to or cancel the dialog box before continuing with the application.

Dialog Type	OSF/Motif	Windows	OS/2	Macintosh
Information				
Question				
Warning				
Critical				

The operating systems shown above have four levels of message dialog boxes, which are symbolized with the icons shown. The first level is the Information Dialog, and is used to convey a message, such as the completion of a task, to the user. The next level is the Question Dialog, and is used to ask the user a question and receive the response. One common question is whether or not to continue with a chosen action. The next level is the Warning Dialog, and is used to confirm that the user wishes to carry an action that is potentially destructive. Two common examples are confirming whether or not to delete or overwrite a file or reformat a diskette. The final level is the Critical Dialog, which informs the user that a problem has occurred which prevents the application from going any further. An example would be the loss of a network drive while an application is using it. All of these message boxes have push buttons at the bottom, one of which is usually Cancel, with the remaining buttons depending on the application and the message being displayed.

Other operating systems also have message dialog boxes, but they are not subdivided into the levels shown above. They may or may not also have icons, depending on the operating system.

Future Directions

The NeXT Computer System™ demonstrated much of the state-of-the-art in Graphical User Interfaces and hardware. The NeXTSTEP operating system's object-oriented interface allows the user to create new applications by modifying and recombining existing program objects. The 3-D, shiny metallic look to the interface is used throughout. The NeXT Computer System can display up to 16 million colors, the maximum on most platforms. The display as well as the print output is generated using PostScript®. Finally, the NeXTSTEP 3.1 operating system has been ported to 486 platforms; allowing cross-platform and platform independent programming, just like the SAS System. All of these features are showing up in the graphical operating systems and interfaces that were previously shown.

All graphical user interfaces, along with the SAS System, are becoming more object-oriented. This means that instead of working with files and applications, you work with objects, represented as icons, and windows. You perform actions by dragging and dropping an icon; and view or change the object's properties by right-clicking on it. IBM's OS/2 Workplace Shell is

object-oriented; and adding a program that uses the Workplace Shell means that the program adds another object or objects (such as in boxes and out boxes if it is an E-mail program) to the Workplace Shell. Once it is added, all applications have access to the new objects using drag and drop. For a more detailed discussion of Object-Oriented User Interfaces, see Theo Mandel's book "The GUI-OOUI War, Windows vs OS: the designer's guide to human-computer interfaces," listed in the References. Both the Windows and Macintosh interfaces will also become more object-oriented, using OLE 2.0 and OpenDoc.

Graphical user interfaces and the SAS System are also using Multimedia to a greater degree. Multimedia includes access to CD-ROM, video, and sound recording and playback. Graphical user interfaces are also showing up on hand held computers or PDAs, complete with pen and handwriting support.

Graphical Applications on the IBM Mainframe

If your current computer network is still IBM Mainframe based, you can still provide graphical SAS applications for your users on the IBM Mainframe. If you are accessing the mainframe through a terminal, and the corporate atmosphere is still hostile towards PCs, then you may want to upgrade your present terminals or acquire replacement terminals. A terminal may also give you faster access; making it good for application development or running SAS/ASSIST.

The following devices support the use of a mouse:

- IBM 3179G with the IBM 3979 expansion unit and an optical mouse.
- IBM 3192G with the IBM 3979 expansion unit and an optical mouse.
- IBM 3472G with a 2-button PS/2 mechanical mouse.

All of these devices offer excellent speed through a local connect. If you are remotely connected, then you may want to request a trial to see if it will be fast enough to be usable.

If you currently own a PC or Workstation, or are considering purchasing one, then you may want to invest in software terminal emulation. You still need to connect your PC or workstation to the mainframe. The terminal emulation software must support Vector Graphics, or emulate one of the above terminals. Check with SAS Institute on which software is recommended, and have your vendor do a test installation to insure that it works correctly.

CONCLUSION

With graphical user interfaces and SAS software version 6.07 and greater now available for all operating platforms, it is possible to write applications that will run on any platform. The graphical user interfaces that are available under Microsoft Windows, OS/2, UNIX, IBM Mainframe, and Macintosh are reasonably consistent, allowing one application to be developed for all these platforms that is easy to use and consistent with what is currently available. (Knowledge of GUI standards will be a necessity!) One version of each application for each platform you support will be needed. However, each version will have 90% or greater of the source code in common. By grouping together all Filename and Libname allocations, DEVICE and GDEVICE statements, and X calls as much as possible, moving between platforms will be made easier. Since all graphical user interfaces share the same basic components, different versions of SAS applications can share common features, and on-line help using SAS/AF CBT screens.

APPENDIX A

The following source code is used to generate the PMENU MAINMENU.PMENU, an action bar menu for a CBT or MENU entry. Note the use of mnemonics, separators, and accelerators.

```

proc pmenu catalog=sas606.example;
menu mainmenu;
  item 'File' menu=filemenu mnemonic='F';
  item 'Options' menu=runmenu mnemonic='O';
  item 'Globals' menu=globemen mnemonic='G';
  item 'Help' menu=helpmenu mnemonic='H';

menu filemenu;
  item 'Open' selection=edit mnemonic='O' accelerate='F2';
  item 'Browse' selection=browse mnemonic='B';
  item 'Print' selection=print mnemonic='P' accelerate='F4';
  separator;
  item 'Exit' dialog=leave mnemonic='x' accelerate='F3';
  selection edit '2';
  selection browse '8';
  selection print '4';
  dialog leave 'bye';
  text #1 @4 ' ';
  text #2 @4 'Are you sure you want to exit the';
  text #3 @4 '(Application Name) Program? ';

menu runmenu;
  (report, graph options listed here.)

menu globemen;
  item 'Menu Off' selection=menuoff mnemonic='M';
  item 'ASSIST' selection=assist mnemonic='A';
  item 'Calculator' mnemonic='C';
  item 'Calendar' selection=calendar mnemonic='d';
  item 'Notepad' mnemonic='N';
  item 'Libname' mnemonic='L';
  item 'Host Command' dialog=host mnemonic='H';
  selection menuoff 'PMENU';
  selection assist '99';
  selection calendar 'appointment';
  dialog host 'X @1';
  text #1 @4 ' ';
  text #2 @4 'Enter the Host command: ';
  text #3 @4 len=15;
  text #5 @4 'NOTE: If you do not specify a command, ';
  text #6 @4 'you will go to the host operating system.';

menu helpmenu;
  item 'Index' selection=help mnemonic='I' accelerate='F1';
  item 'Using Help' selection=helphelp mnemonic='U';
  item 'SAS System Help' selection=sashelp mnemonic='S';
  separator;
  item 'About (Application Name)' selection=about
    mnemonic='A';
  selection help 'help';
  selection helphelp 'helphelp';
  selection sashelp 'sashelp';
  selection about 'about';

```

REFERENCES/FOR FURTHER READING

USER INTERFACE DESIGN

Professional SAS User Interfaces (#C56197). SAS Institute Author Service.

Mandel, Theo (1994). *The GUI-OOUI War, Windows vs OS: the designer's guide to human-computer interfaces*. Van Nostrand Reinhold.

Tognazzini, Bruce "TOG" (1992). *TOG on Interface*. Addison-Wesley Publishing Co.

Moncrief, Frank (February/March 1993). "Master the Principals of User Interface Design," *BASICPro*, pp. 20-23.

Hefner, James (1993). "How to Implement a Graphical User Interface Using the SAS® System, Release 6.06 under MVS (without Confusing Your Present Users)," *Proceedings of the*

Eighteenth Annual SAS Users Group International Conference, pp. 645-654.

Human Factors in Computer Systems: CHI '92 Conference Proceedings. Addison-Wesley Publishing Co.

Heckel, Paul (1991). *The Elements of Friendly Software Design*. Warner Books (also published by Sybex).

The Art of Human-Computer Interface Design. Addison-Wesley Publishing Co.

Norman, Donald A. (1990). *The Design of Everyday Things*. Doubleday.

Metaphors We Live By. The University of Chicago Press.

USER INTERFACE STANDARDS

A SAS Companion for each operating system that needs support.

OSF/Motif
Open Software Foundation®. *OSF/Motif Style Guide*. Prentice Hall Publishing Co.

Microsoft Windows
Microsoft Corporation (1992). *The Windows Interface: An Application Design Guide*. Microsoft Press.

IBM OS/2
IBM Corporation (1992). *Object-Oriented User Interface Design Common User Access™ Guidelines*. Que Corporation.

Apple Macintosh
Apple Computer, Inc. *Apple Human Interface Guidelines*.

IBM® Mainframe
IBM Corporation (1992). *Systems Application Architecture Common User Access Basic Interface Design Guide 1991 Addenda (GG22-9508)*. IBM Corporation.

CONTACT:

James D. Hefner
3105 West Lucas
Beaumont, TX 77706
(409)899-2127
InterNet: Hefner@AOL.COM

Engineer III
Entergy Corporation
Technology Development-
Modeling & Systems Engineering Group
Mail Stop T-EP-7
P.O. Box 2951
Beaumont, TX 77704
(409)838-6631 ext. 2662

"Modeling & Systems Engineering are value driven, customer oriented consultants supporting Fossil Operations in optimizing the effectiveness of current and future assets by providing technical specialty information management and thermal performance, reliability/availability/maintainability, and economic analysis consulting services."

SAS, SAS/AF, SAS/ASSIST, SAS/EIS, SAS/GRAPH, SAS/FSP and SAS/CONNECT are registered trademarks or trademarks of SAS Institute, Inc. in the USA and other countries. AIX, Common User Access, CUA, IBM, OS/2, and PC DOS are registered trademarks or trademarks of International Business Machines Corporation. Amiga and Commodore-Amiga are registered trademarks or trademarks of Commodore International Limited. ® indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.

My thanks to the Lord Jesus Christ for all he has done . . .
By a new and living way, which he hath consecrated for us, ...

Hebrews 10:20.