THE SAS SYSTEM AND THE UNIX ENVIRONMENT:
YOU GOTTA LUV' IT

David Wooldridge, Hewlett-Packard Company
Claire Castell, Sun Microsystems, Inc.

The SAS® System has been running on various Unix® platforms for a couple of years. With the imminent release of version 6.07, it will run on more Unix platforms and under X Windows. Windows provides an ease of use that is unmatched.

One advantage of Unix is that it runs on large systems in typically housed in data-centers, and on workstations on users’ desks. Workstations are usually setup in a multi-user mode on a network so that multiple users can use the SAS System on datasets on multiple workstations.

X is a windowing system developed at MIT and provides a common windowing system that can be used across many systems and displays. X Windows provides the capability of performing operations on multiple systems while having the displays from those multiple processes on multiple system as windows on one local workstation.

This tutorial will introduce you to using the SAS System in the X Windows environment under Unix. We will be working with version 6.07 on a number of different platforms.

X is the lower level protocol that provides the windows themselves. The higher level functions the ability to move, resize, deal with icons, menuing, the “look and feel”, is handled by the window manager and the widget set used by the programmers of the application. Unfortunately, a number of different window managers and widget sets have been developed, and can be incompatible. Fortunately however, the industry has recognized the need for standards in this area. Two competing standards have evolved: Motif, supported by HP, DEC, IBM, and a number of others; and OpenLook supported by AT&T and Sun. Both are represented in this tutorial.

Using a Mouse.

The benefits of using X Windows system are fully realized with the use of a mouse. On your display is a mouse pointer (sprite, cursor). Moving the mouse causes the pointer to move in a corresponding manner. The mouse pointer is often imprecisely referred to as the mouse. We will sometimes slip into this idiomatic usage, although we usually use pointer.

A mouse, on a system running X, typically has two or three buttons which you can press to cause actions in the window system. Two button mice can often get the same results as the third button on a three button mouse by pressing both buttons simultaneously. Buttons are referred to as left, right, and middle.

Pressing a mouse button is often referred to as clicking, after the sound it makes.

There are a number of button press actions used in X Windows and in the SAS System. The most common of which is the simple button press. Button in this sense is a box on the screen often designed to look like a button on a hardware panel. Put the pointer into one of these “buttons” and press and release a mouse button. Another action “click and drag”. Place the pointer on some widget on the screen, such as a resize handle, pressing a mouse button and moving the pointer without releasing the mouse button. The widget will move with the mouse pointer. This is used for moving or resizing windows, or with a scroll bar, for example.

Different windows and fields have different actions defined for the button presses, depending upon the application. Pressing the left button can have one action, while pressing the right button can have another. Some applications or windows define modifier keys so that holding one of the modifier keys while pressing a mouse button causes yet another action. There are several different commonly used modifier keys: crtl, shift, or ext (or alt).

Another mouse action is the double click. This involves pressing a mouse button twice in rapid succession.

Active Window and Keyboard focus

Under normal usage, whenever the mouse pointer is in a window, that window is “active” and has the “keyboard focus”. The borders of the window usually change to indicate the active window. Any input from the keyboard will be acted upon by that window.

There may be different entry or edit fields within a window. Only one of these will have the keyboard focus at a time. Sometimes moving the mouse into the field will activate it and allow you to enter or edit values. Sometimes you must move the mouse into a field and click a mouse button. Sometimes you can use the tab key to switch the focus between the fields. The usual way to determine which field has the focus is to see which field contains a text cursor.

Raise, Lower, and Iconify

Most window managers support overlapping windows. A window may partially or completely obscure a window behind it. The windows can be visualized like pieces of paper stacked or scattered on a desk. The process of taking one of the windows from somewhere in the stack and putting it on top is called
"raising" it. Taking a window and putting it on the bottom is called "lowering" it. X Windows have another nice feature the ability to collapse a window into its icon. This may be analogous to putting it into a file drawer. Every major window on the screen has a corresponding icon elsewhere on the screen. When you iconify (minimize) the window, it disappears from the screen and its icon changes appearance somehow. You may get into the window back by clicking on its icon.

Push buttons

The most common window action is the button press, which causes some application actions to happen. Often these "push-buttons" appear as labeled buttons on the screen. In 3-D windows, they seem to stand out or set back from the screen. Pressing the button "arms" it, causing some visual indication that it has been pressed such as an appearance of being pushed in. Releasing the button causes the button's action to occur. Moving the mouse out of the window without releasing the button disarms it and no actions occurs. On other types of buttons, the actions can occur immediately upon the button press.

Pull down menus

Some buttons, especially those on a "menu-bar" cause a pull down menu to appear. Menu bars are usually situated at the top of a window, but SAS has windows with a menu bar at the bottom also. Pressing a menu-bar button causes a pull-down menu to appear. Releasing the button causes it to disappear. Dragging the mouse pointer through the selections in the pull-down menu "arms" each item as it is traversed. Releasing the mouse button while the mouse pointer is in a menu item selects that item. Sometimes menu items cascade. Cascading items usually have an arrow to the right indicating more menus. Moving the mouse pointer to the right in one of these causes the next menu to appear. Moving the cursor out of the cascading menu causes to disappear.

Pop-up menus

Pop-up menus are very similar to pull-down menus except that they appear after some action other than a menu-bar selection. For example, the screen's root window usually has a pop-up menu that appears when you depress a mouse button while in the root window.

Toggle buttons

Toggle buttons are buttons, usually with some attached text, that are used for selecting whether some application option is on or off. The button has an indication that it is on, it is usually filled with some color. Clicking on the button turns it on. Clicking again turns it off. You may click on the button itself, or on its label.

Radio buttons

Radio buttons are a group of two or more toggle buttons with the behavior that only one may be selected at a time. Clicking on one of them causes any other that has been selected to be deselected. It works like car radio selection buttons. When one is pushed in, others pop out.

Selection lists

A selection list is a window that may contain a number of selections. Depending on the application, you may select one or more selections from the list. Selecting one of the items causes it to be highlighted. You select an item by clicking a mouse button on it. If the application permits, you may make multiple selections by clicking on an item and dragging through several others.

Exercise 1: Using Windows

The SAS System is already running on your workstations. The first exercise is to become comfortable with using the mouse, then to practice the basic windows functions moving and resizing windows, using the scroll bar, and using the pull-down menus.

Exercise 2: The SAS System Display Manager

Throughout this paper, actions for you to take are indented. Explanatory material is in normal paragraph form.

The first thing we will do is to bring a SAS program into the SAS Program Editor and submit it. This program reads a flat file containing revenue and expense information for the Mardi Gras Bead Company. A SAS dataset is created that we will use for future exercises. The program will also generate a printout of the data. The program is provided as an appendix to this paper.

```
Select File from the Program Editor pull-down menu
Select Open
Select Read File
Select ~ssuser/Pyms/Bead1.sas
Push the OK button
```

The program appears in the SAS program editor. Grow the Program Editor window so that the program is easier to read. Scroll through the program to see what it does. Now submit the program.

```
Select the Locals pull-down
Select Submit
```

The program will disappear from the Program Editor. You will see the SAS log scrolling in the Log window. You will see the Proc Print output appear in the Output window.

```
Move to the Output window
Scroll through the output
```
Select the Edit pull-down
Select Clear Text
Move to the Log window
Scroll through the log to see how the job processed
Move to the Program Editor window
Select the Locals pull-down
Select Recall Text
Add your name to the Title
Submit the program as before

Using SAS/Assist® Software in Version 6.07

Types of Windows in SAS/Assist

Block Menus Used to select major type of activities by clicking on the icon.

CBT Windows Used to offer help on different topics.

Form Dialogs Used to completely define a problem for the SAS system to solve. Complete by filling out appropriate blanks in the form.

Pop-up Dialogs Used to provide information to be selected by means of character input, choosing an item from a selection list, turning a toggle button on or off, or selecting from a radio button.

Exercise 3: Create and Store Graphs with SAS/ASSIST

We will generate two high-resolution graphs from the data read with the first exercise. The first is a pie chart showing revenue by product. The second is a stacked bar chart showing revenue by month with the bar stacking indicating distribution channels.

Start SAS/ASSIST from the program editor

Select the Globals pull-down
Select ASSIST

You now see the SAS/ASSIST main block menu.

Select Graphics
Select High-Resolution from the dialog box

You now see the graphics block menu.

Select Pie Charts

This brings up a form dialog for setting up the pie chart. When an ASSIST dialog form is displayed, SAS makes some assumptions by selecting defaults. It shows you its defaults, allowing you to change them. If it can make no reasonable assumptions, and the value is required, SAS will indicate with a value of REQUIRED- displaying in the field. You must select these fields and choose a value.

Select Active Dataset

This brings up a list dialog with all of the data sets that SAS knows about.

Select the dataset MGB607_1

We are now back at the Pie form

Select Chart Variable

This brings up a list dialog with all of the variables in the active data set.

Select PRODUCT

Back at the form Dialog

Select Slice Values

This brings up a form dialog with a radio box to select the statistic to be charted and a selection for the analysis variable.

Select Sum for the statistic
Select Analysis Variable

This will bring up a list of variables appropriate for the kind of statistic selected.

Select REVENUE
Push the OK button

We have now defined the problem we want to graph.

Push the RUN button

SAS/Assist generates a SAS program when you run a problem. It is regular SAS code that can be saved and reused in any way desired, or modified and enhanced by a SAS programmer. We want to save this program for later use.

Press the Results button

This brings up a menu box.

Select Save Current Program

A dialog box will appear

Enter "PIE"
Press the OK button
A message box will appear stating that the program has been stored. It is being stored as a "source" entry in a SAS catalog. Enter goback, and OK until back at the Graphics Block menu.

Now we are going to create the bar chart.

Select Bar Charts
Select Vertical Bar Charts
Select Stack bar charts by clicking on the highlighted word, Stacked

You will now have the stacked bar chart dialog form.

Select Bar Values
Choose Sum as Statistic to be charted
Select Analysis Variable
Choose REVENUE
Push OK
Select Stacking Variable
Choose CHANNEL
Select Chart Variable
Choose MNOTH
Select Additional Options
Choose Number of bars
Choose Use each discrete chart variable value
Push Goback until back at Stacked bar form
Push RUN

Now save this program

Push RESULTS
Choose Save current program
Enter "bar"

This brings up the Insight main menu window. You may want to clean up your display by minimizing most other windows as Insight uses a lot of display real estate itself.

First we must choose a data set to analyze

Select Open from the main menu
Choose SASUSER library
Choose FITNESS dataset

This will bring up a window with the dataset displayed in tabular form.

Enlarge the window to display a usable amount of the data

There are two buttons on the left side of this window. The top one is used to control window options. The bottom one controls data options.

We want the data sorted by age,

Select the AGE variable by clicking the top of the column labelled AGE
Select the data (bottom) button
Choose Sort

Now we want to color code the observations by age group. First get some necessary tools.

Select Palettes from the main menu
Choose Color
Select Palettes
Choose Tools

Now make your selection.

Pick the select tool (arrow) from the tools menu
Click and drag to select all 30 year olds

Clicking and dragging with the select tool will create a dotted line box to help indicate the observations being selected. Drag the box through the observation numbers column, picking all those observations with 30 year olds. Releasing the button activates your selections and highlights the observations selected.

Click on a Yellow color in the color tool.

The markers for the observations selected will turn yellow

You may clear any active selections by clicking in the upper left corner of the window. The highlighting will go away, but the colored markers will remain.

Clear selections
Select 40 year olds
Pick a green color

The Power of SAS/Insight® in Version 6.07

There are different types of windows used in SAS/Insight Data windows, graph windows, and analysis windows. SAS/Insight has a Main menu which is brought up as its own window with a number of buttons to create new windows or dialogs, or to modify existing windows. Palettes in SAS/Insight provide tools for its operations. The tools palette provides the select tool, the move tool, and the focus tool. The color palette supplies a tool to color code observations. The markers palette changes the shapes of the markers that identify observations.

Exercise 4: Analyze Your Data with SAS/Insight

Run SAS/Insight by either entering the command "insight" on the command line, or enter and submit the program "proc insight; run;".
You may use the scroll bar to scroll to observations that may be out of the window.

Clear selections
Select 50 year olds
Pick a blue color

Now lets make a bar chart of some of the data.

Select Graphs from the main menu
Select Bar Chart (Y)

A bar chart dialog will appear

Select AGE
Push Y button
Push OK button

This will bring up a bar chart

Enlarge the window

Enlarging the window does not enlarge the bar chart, but merely gives you more room to work with. You enlarge the bar chart itself by using the "focus" tool from the tools menu (the bottom tool). Put the focus tool in the bar window and click to enlarge or reduce the chart. Notice that the tool itself changes size as it is moved around the window.

Highlight the 50 year olds

With the select tool go back to the tabular data and select the 50 year olds. Note that the 50 year olds in the bar area also highlights.

Clear the selection

Go straight to the sexy stuff

Select Graphs from the main menu
Select Rotating Plot
Enlarge the window
Select AGE
Push Z button
Select RUNTIME
Push Y button
Select MAXPULSE
Push X button
Push OK button

You now have a scattergram in 3 dimensions.

You will see the different groups in the colors you selected. With the selection tool, pick a single point. Notice the observation highlights in the tabular data. Select a point on the data and it is highlighted in the scattergram. Select a bar in the bar chart, and it is highlighted in the scattergram and data.

Play a little

Select the rotating plot button (lower left button on the plot) and select rays, box, or depth cueing.
Select the "Hand" tool and give the plot a push with the hand--swat at it, and watch it spin.

Using Proc REPORT

Proc Report is an procedure that you can use to interactively define how you want a report to look, then you can store the report definition and print datasets against it.

Defining the report

Run the following program from the program editor

options nodate nonumber ;
proc report fs data=SASUSER.MGB607_1 ; title;
run;

Proc report variable types

Variable types are chosen from the Variable Definition dialog when you have selected Define from the Edit menu

Display (default) Each row displays a value for each observation.

Order Displays the value of each order only once. (Like Proc Print with BY and ID statements)

Group Collapses into one row all observations from the data set that share a unique combination of values. (Like Proc Summary)

Across Similar to group variables, but displays the groups it creates horizontally rather than vertically. (Like Proc Transpose)
Analysis

These are used to group or order variables to calculate a statistic. You must specify a statistic in the Statistic field of the definition window.

Computed

Variables not found in the data set but computed by the report. Created from one of the ADDING windows

Exercise 5: Proc Report

We will make two reports. The first is a report of profit for the Mardi Gras Bead Company. The Second is a report of revenue and profit margins, summarized.

Many of the pull-down menu selections have an accelerator listed in a table at the bottom of the window. You can either use the listed function key, or simply click on the entry in the table.

Start by making the column wider for "channel" with the CGrow command under the Edit pull-down.

Select the channel variable by clicking on the column heading channel
Select CGrow
Select month variable
Select Define
Enter format of mmddyy9.
Select Order variable

Define product and channel as Order variables.

Define revenue and cost as display variables with format=dollar9.

Create a computed variable named profit

Select cost variable
Select Add_right from the Edit menu
Select computed variable
Enter “profit” as variable name
Select Edit program
Enter the program

\[ \text{profit} = \text{revenue} - \text{cost} \]
Select End from File menu
Push OK

Now let's modify this report and make the second report.

Delete month and profit by selecting each column and selecting Delete from the Edit menu.
Define product and channel as group variables.
Define revenue and cost as analysis variables, and assign statistic to be sum
Define a new computed variable called margin to the right of cost. Enter the program:

\[ \text{margin} = (\text{revenue.sum} - \text{cost.sum})/\text{revenue.sum} \]

Note the suffix. If the statistic assigned was mean, you would enter revenue.mean.

Create sub-totals by product
Select product
Select break from the edit menu

A location window will pop up
Select after detail
The Break window will pop up.
Select overline
Select skip
Select summarize
Push OK

Create the grand totals for the report.
Select RBreak and the Break window will pop up.
Select Double Overline
Select Double Underline
Select Summarize
Select Color
Push OK

Save report definition
Select RStore from the File Menu.
Enter sasuser.reports.report2

Add some titles by selecting Titles from the Globals menu.
Select Refresh from the Edit menu to implement the titles on this report.

Change the column heading for month to "Period". Just type over the label.
Running the following program will create this report and mail a report to a manager in the Mardi Gras Bead Company.

```
opions nodate nonumber;
filename mail pipe 'mailx -8 Report manager-id';
proc printto print=mail;
proc report data=sasuser.beads
   report=sasuser.reports.report2
title 'Mardi Gras Bead Company';
run;
proc printto; run;
```

EIS: Executive Information System

SAS/Assist provides an EIS facility that can be used to construct your own SAS/Assist style menus for use within your own company.

EIS is run from SAS/Assist, and uses the same types of menus and dialogs for interacting with the user or programmer. Use the EIS selection from the Assist main menu to start EIS.

The first dialog asks whether you want to run a Public Application, a Private Application, or build an EIS application. Public applications are those that the SAS Institute or a programmer at your company has put together for anyone to use. Private applications are those which you or your work group have put together.

You will be building entries in an EIS dataset. Each menu or menu selection or operation consists of a separate entry in the dataset. Your main menu will be an entry. Each operation or sub-menu that each main menu selection calls will be an entry.

**Exercise 6: Create a New Menu in EIS**

We will create a private application that has a block menu with three buttons. The first one is labeled Pie, contains an appropriate picture, and runs a SAS program to generate the same type of pie chart as the Assist exercise against the beads dataset. (We use the source saved from the Assist exercise.) The second button is labeled Bar, and will run the same stacked bar chart as the Assist example. The third button is labeled Goback, and exits your private application.

Select **ASSIST** from the Globals pull-down
Select **EIS**
Select **Build EIS**
Select **Create/Update Applications**

You will now be in a list menu of all of the elements of your EIS. At this time, it is blank. Start by adding the first element, the main menu.

Push the ADD button

You will now see list of different kinds of EIS elements you can create. We will start with our main menu.

Select **Block Menu**

We are now in a form dialog which will allow us to define the characteristics of our menu. We will give it the name mgbmain, and the title MGBCo Executive Info. This title will appear as the title at the top of the menu.

Enter name "mgbmain"
Enter desc "MGBCo Executive Info" Select Use Icons
Select Color
Use Prev and Next to choose a color
Click the check button when ok
Select **Edit Menu Text**

We are now in a form dialog from which we can define the selections of our menu.

Enter name "rept1"
Enter desc "Pie Report" Select **Execute SAS Program**
Select **Icon**
Select **Standard Icons** from dialog box

We will now be presented with icons that SAS has provided for our use. Use the scroll bar to scroll through the list. We like icon 82.

Select icon 82

Now use the scroll bar to scroll down to the next form. This will be the next button on our menu. You can scroll back and forth through the buttons already defined, and make any changes. Scrolling down from the last defined entry adds a new one.

Scroll down
Enter name "rept2"
Enter desc "Bar Report" Choose **Execute SAS Program**
Select **Icon**
Select **Standard Icons** Select icon 81

Scroll down
Enter name "goback"
Enter desc "Go Back" Choose **Exit Block**
Select **Icon**
Select **Standard Icons** Select icon 69
Use the OK and Goback buttons to get back to the list menu that we first saw after Build EIS. Note that it now contains one element, the block menu we just created.

We must now create elements to execute the actions of the menu selections we created in our main menu. At this time they are simply buttons on a menu which do nothing (except the goback which does already work).

Push the Add button
Choose Execute a SAS program
Enter name "rept1"

This must be the same name as we used above. The menu button in the main menu will look for an element here with that name and run it.

Select Source in Catalog
Select library SASUSER
Select catalog SASPGMS
Select object PIE

This pie is the source for the pie chart that we saved earlier in the SAS/Assist exercise.

Enter OK and goback until you reach the EIS list again.

Push the Add button
Choose Execute a SAS program
Enter name "rept2"
Select Source in Catalog
Select library SASUSER
Select catalog SASPGMS
Select object BAR.

Enter OK and goback until you reach the EIS list again.

We have now defined our EIS. We can test execute it now by selecting the mbmain item, then selecting the test button. All of our buttons now work and will generate reports or charts from our beads dataset. We would execute it later by selecting Run Private Applications from SAS/Assist's EIS selection.

Appendix

Bead1.sas

```sas
/*==========================*/
/* ~/sasuser/Pgms/Bead1.sas */
/*==========================*/
filename in '~/sasuser/MGBeadCo';
libname out '~/sasuser/';

data out.mgb607_1;
infile in;
input @1 month mmddyy8.
@10 product $char7.
@20 channel $char6.
@29 revenue 7.
@30 cost 6.
run;
proc print;
  format month mmddyy8.
  revenue cost commall.
run;
/* Note: To request a copy of the raw */
/* data that was used in the tutorial, */
/* send an e-mail request to one of */
/* the authors. */
```

Authors

David Wooldridge
Hewlett-Packard Company
3000 Hanover St., 20CF
Palo Alto, CA 94304
(415) 857-4265
wooldrid@corp.hp.com

Claire Castell
Sun Microsystems, Inc.
MIL03-73
2550 Garcia Ave.
Mountain View, CA 94043
(408) 276-4066
clairec@EBay.Sun.COM

SAS is a registered trademark of SAS Institute, Inc.
Unix is a registered trademark of AT&T
SAS/ASSIST is a registered trademark of SAS Institute, Inc.
SAS/Insight is a registered trademark of SAS Institute, Inc.