SAVING TIME WITH THE SAS DISPLAY MANAGER

Phil Busby, SAS Institute Inc.

The SAS® display manager is a full-screen control program that greatly enhances the productivity of the SAS programmer. It is now part of the SAS System for minicomputers, running in full-screen mode. This paper describes the display manager and shows how you can use it to create and run SAS programs quickly and effectively.

First, a discussion of the basic functions of the display manager and instructions for controlling the program, log, output, and help windows when running a simple SAS program are presented. Then the powerful editing commands are described, followed by a description of each screen command that you can enter on the command line. The final part of this paper illustrates how the program function keys can be used for the fast entry of commands.

Begin the SAS session by entering the following command:

```
SAS/FSD=TEK410S
```  

Here, separate the command keyword SAS from the options you want set at initialization time with a slash, and in this case, the only option needed is the full-screen device option. This specification says two things to the SAS System: first, you want full-screen mode, and second, your full-screen device is a Tektronix 4105 terminal. By specifying what kind of terminal you are using, the SAS System can choose the correct device driver to use, and it knows how many rows and columns are available for display.

As you go into full-screen mode, the SOURCE option is turned on so that your SAS source lines are written to the log window.

Here is the initial screen you see when the display manager takes control. It is really two logically distinct displays: the program editor window, below, is where SAS source statements are entered, and the log window, above, is where the SAS log lines are displayed, showing the results of your executed SAS source statements.

To see exactly how each window works, enter a trivial example. Key in the following:

```
DATA A;
X=1;
RUN;
```  

and enter the SUBMIT command. The source statements appear on the log window, followed by the note, which tells you that your data set was created successfully.

Notice that the program editor lines are blank, ready for new source input.

```
Command mode  SAS Log 10:48

Copyright by 1985 SAS Institute Inc., Cary, N.C. 27511. All rights reserved. This version of SAS Release 4.05 is for INSTITUTE EDITION, only.

NOTE: THE DATA SET WORKS A HAS 2 OBSERVATIONS AND 2 VARIABIES.

```

Now you can enter

```
PROC PRINT;
RUN;
```  

and submit it to the SAS compiler, which causes the PRINT procedure to print data set A. PROC PRINT produces a page of printed output and invokes the display manager, which displays the page. The display manager also sounds the alarm, which indicates that the procedure is complete, and the last page is displayed. Now, simply press ENTER to return to the log/program screen.

SAS is the registered trademark of SAS Institute Inc., Cary, NC, USA.
To complete this example, you need to examine two more features of the SAS System. First, to run a graphics procedure, key in

```
PROC GCHART;
PIE X;
RUN;
```

and enter the SUBMIT command, which causes the screen to be cleared and allows the GCHART procedure to take over, displaying a pie chart of the variable X.

After examining the chart and returning to the log and program windows, go up to the program editor command line and enter the HELP command.

The HELP command causes the log window to be replaced by the SAS help window so you can browse the requested information while typing SAS statements into the program window. The help window remains displayed until you enter the SUBMIT or LOG command, which recalls the log window.

Now that you have seen how to move from screen to screen, go over the different commands available on each window. First, split the screen to make more room for program lines by moving the cursor where you want the dashed line to be and pressing the SPLIT key (program function key 9). Then, remove the line numbers from the program editor window with the NUMBER OFF command, allowing fast entry of SAS statements. Type the following:

```
DATA E;
DO J=1 to 5;
  OUTPUT;
END;
```

Before editing these lines, put the line numbers back on the program editor window with the NUMBER ON command.
The following seven line commands operate on a single line:

- \( \text{n} \) shift right
- \( \text{<n} \) shift left
- \( \text{In} \) insert
- \( \text{Rn} \) repeat
- \( \text{Dn} \) delete
- \( \text{M} \) move
- \( \text{C} \) copy.

Except for \( \text{M} \) and \( \text{C} \), all of these commands can be followed by an integer operand to specify how many times to do the operation. For example, suppose you decide to shift the RUN statement that you entered above to the right to create an indentation. Simply enter \( >4 \) in the line number field and press ENTER. The display manager moves the data on that line four columns to the right. It knows to indent 4 spaces, not 4005, because it scans the line number field backwards to extract the integer operand, and it ignores digits that match the previously displayed line number digits. In this example, if you had wanted to indent 40 columns, it would have been necessary to blank out the zero after the >40.

If you try to shift the text too far, you receive the invalid line command message; the alarm sounds; the cursor is placed on the line command field; and the line command field attribute is changed to reverse video. The same thing happens with any invalid line command, for instance QQQ, and the erroneous command must be erased or fixed before continuing.
Now you can insert a new line. Simply type an I on the line before the place the new line should go and press the ENTER key. The screen comes back with the cursor placed in the first column of the new line's data area.

Notice that the new line was initialized to blanks. For another initial value for inserts, use the MASK command. For example, if you have multiple comment lines to put in, you can type the MASK command on a line with comment delimiters and press the ENTER key. The display manager extracts the mask and throws the line away, but the next time you insert lines, they will be initialized to the mask value.
To change the mask, just enter another MASK command. For example, set the mask back to blanks by typing MASK on a blank line.

Inside the DO loop type the statement K=J;, and for that line enter the line command R2 to repeat the line twice. Now you can alter the statement to read L=J on one line and M=J on the other line by changing just one character on the repeated lines.

You can put a line command on more than one line at a time before pressing the ENTER key, and all the commands will be processed by the program editor. Add a new line after the DATA statement with an I on that line and delete the line that has M=J by keying a D line command there. Both operations are done when you press ENTER.
There are two more line commands that operate on single data lines: M and C. The M line command moves the specified source line to the destination you indicate with a B (Before) or an A (After). For example, you can reverse the order of two lines by moving the first after the second.

Whereas the single line commands are entered on one data line, the double line commands operate on a group of lines. The double line commands are

- `>`n shift right
- `<<`n shift left
- DD delete
- RRn repeat
- CC copy
- MM move.

The double line commands work only in pairs. You can delete a block of lines by keying DD on the first line and DD on the last line in the block, or you can repeat an entire block by typing RR on the first line and RR on the last line in the block and pressing ENTER.
The shift right, shift left, and repeat double commands are the only block line commands allowing an optional integer operand to follow the command. For example, you can indent the rest of the statements in this DATA step by typing ">>3" on the first line in the block to be shifted and by typing ">>" on the last line in the block. When you press ENTER, all the lines are shifted right three columns.

Consider another example. If you type "CC" on the first line command field and press ENTER, the "CC" stays there and nothing happens. Type "CC" on another line to designate the block and press ENTER. Again, nothing happens, and the block copy is still pending because you did not specify where to put the copied lines. If you type an "A" on the line command field of the RUN statement and press ENTER, the copied lines appear after the RUN statement.
Now you can delete these extra lines and leave the DATA step ready to execute.

Now send the SAS source statement lines to the SAS compiler with the SUBMIT command.
When the screen returns, your results appear in the log window, and the program window is blanked out and ready for more SAS statements.

Now move to the log window and browse the log. You cannot change the data in the log window, but you can alter the way they are displayed with several screen commands that you enter on the log command line. The commands recognized on the log window are the color setting commands, scroll commands, FIND, BYE, RULE, PRINT, X, ASSIGN, SAVE, and GO. These commands are recognized on every display manager window; that is, every command you can use on the log command line is also allowed on the program, output, and help command lines.

The color setting commands allow you to change the color and attribute of four kinds of displayed data: the banner, the source lines on the log, the protected fields, and the unprotected fields. Banner fields include the dashed line, the word command, the window name, and, if present, the line numbers. To change the log banner lines to the color cyan with the underline attribute, enter CBANNER CU.

The log source lines will change to white when you enter CSOURCE W.

The other protected lines on the log change to white with the blinking attribute by entering the CPROT W B command.

Then you can change the lines back to yellow with normal attributes by entering CPROT Y.

If you want the unprotected command line itself to be purple with reverse video, enter CUNPROT P R.
If your terminal does not have color, you can still change the attribute. For example, to highlight the command field, enter CUPROT W H.

The scroll commands are FORWARD, BACKWARD, TOP, BOTTOM, LEFT, and RIGHT. If you enter the FORWARD command on the log command line, the displayed lines do not change and the alarm sounds to tell you that you cannot scroll forward since the bottom line of the log is already displayed.

BACKWARD causes the previous three log lines to scroll into view. You get a vertical scroll amount of 3 because the default scroll amount is set to HALF, and there are six log lines displayed.

You can change the vertical scroll amount by entering VSCROLL PAGE or some other value, such as VSCROLL 2. The VSCROLL command changes the vertical scroll amount, but it does not actually scroll.
You may temporarily override the scroll amount by typing BACKWARD n, where n is some integer value. You can back up one line by entering BACK 1.

Now, if you enter BACKWARD, you scroll back two lines.

Or you can back up to the beginning of the log by entering the BACKWARD MAX command.
The BOTTOM command causes the last log line to be the bottom line displayed in the window. It is equivalent to entering FORWARD MAX.

Horizontal scrolling is similar to vertical scrolling. The HSCROLL command sets the horizontal scroll amount; the default is HALF the width of your screen. If you enter the RIGHT command, the log lines shift over only one line because you already had columns one through seventy-nine of the log on the screen, and the log data lines are eighty characters long.

You can also make any particular line the top line displayed in the window by typing the line number on the command line and pressing ENTER. For example, to go to line one, type the integer 1 on the command line and press ENTER.

The LEFT M command moves the window position the maximum amount to the left, that is, back to column one.
Now suppose you want to locate a given character string somewhere in the log, say the letter J. Type FIND J on the command line and press ENTER, and the system scans the log lines, looking for J. It finds the DO J=1 TO 5 statement and does an automatic vertical scroll to display that log line since it was not already displayed. The cursor is positioned under the first character of the search string.

You can enter the REPEAT FIND command (or enter FIND *), and the system puts the cursor under the next occurrence of J that it finds, starting its search one character to the right of where it is when you enter the REPEAT FIND command.
If there are no more occurrences of the search argument, then you get the note 'string not found' on the message line.

The FIND command can take an optional operand following the search string: PREFIX, SUFFIX, or WORD. A word is any string that has a delimiter on both sides. A delimiter is anything other than an alphabetic letter or a digit. A prefix is any string that has a delimiter on the left but not on the right, and a suffix has a delimiter on the right but not on the left. There is an assumed delimiter at the start of each line. For example, if you enter FIND PUT SUF, the system finds the suffix PUT in the word OUTPUT.

The BYE command ends your SAS session. The RULE command causes the rule to be displayed on the message line when no other message is pending.

The PRINT command sends your currently displayed data lines to the printer. If you enter the PRINT command here, the note '6 lines printed' appears.
You can print off the entire log by entering PRINT ALL.

This command causes the system to display the date and time on the line following the line that the cursor is on. The display manager is not aware of the fact that the VMS, the host operating system, wrote anything on the screen, and if you enter the REFRESH command, the display manager erases your screen and returns it to the way it was before the X command was executed.

Now suppose you want to copy the log lines to a data set so they will be available after you end your full-screen session. First assign a logical file name to a data set name with the ASSIGN command, such as ASSIGN.MYLOG.DAT LOG and press ENTER.

The X command calls the host operating system and passes to it whatever you type following the X. Type X SHOW TIM and press ENTER.
Then you can copy all the lines in the log to MYLOG.DAT by entering SAVE LOG.

Notice that after you enter the SAVE LOG command, a note indicating that the log was saved appears.

Finally, you need to be familiar with the GO command, which is used to transfer to a new screen without displaying any intermediate screens. Currently the only new screen you can go to is the output window, so type GO OUTPUT LINE to get to the SAS output window containing your printed output. Add the LINE keyword to get a command line and go into line browse mode. Line browse mode is the normal processing mode for the log and help windows. Page browse mode is the default for the output window.

To return to the log/program screen, press the END key, program function key 3.

There are commands you can use on the program editor window only. The RECALL command used on the program editor command line recalls all the SAS source statements typed since the last time you entered the SUBMIT command.
You can continue recalling until you receive the error message 'no prior submit', which indicates that every block of lines submitted since the start of your session has been recalled. After another SUBMIT, the recall function returns the screen to the bottom of the stack of submitted blocks of code.

Enter RECALL again, and the display manager recalls the lines typed since your next-to-last submission, putting them in front of any other lines in the window and causing the rest of the lines to move down.
Another way to get previously executed lines back into the program window is to use the COPY command. Type COPY on the command line followed by the first and last line numbers of the statements as they appear on the log, and those lines are copied into the program window at the bottom.

COPY followed by a logical file name brings in an external file. For example, ASSIGN the data set TEST.SAS to the logical name TEST and follow that command with COPY TEST. The lines from the external data set TEST.SAS are included in the window at the location specified with the A (After) line command. When the logical name is unassigned, the display manager automatically assigns logical name SAS.
The other lines in the window are pushed down, and you can view them by scrolling FORWARD.

Even with OUTPUT OFF, your printed output is still there, and if you want to see it, simply enter GO OUTPUT.

The OUTPUT command is used to control the automatic display of printed output pages when your submitted SAS code contains a PROC step. Type OUTPUT OFF and press ENTER. Now you can execute all of this SAS code, but no printed output is displayed. (Graphic output is not affected.)
Note: on the program editor command line, the OUTPUT command not followed by either ON or OFF causes the log or help window to be replaced by the output window.

When browsing printed output in the page browse mode, you can go from page to page by pressing the program function keys for FORWARD, BACKWARD, TOP, and BOTTOM. TOP takes you to the first page of output for the currently displayed procedure. If you are already on the first page, the TOP program function key displays the first page for the previously executed procedure.

<table>
<thead>
<tr>
<th>Command</th>
<th>SAS Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>GENERAL LINEAR MODELS PROCEDURE</td>
<td>GENERAL LINEAR MODELS PROCEDURE</td>
</tr>
<tr>
<td>DE</td>
<td>SUM OF SQUARES</td>
</tr>
<tr>
<td>1</td>
<td>10.00000000</td>
</tr>
<tr>
<td>2</td>
<td>0.00000000</td>
</tr>
<tr>
<td>3</td>
<td>0.00000000</td>
</tr>
</tbody>
</table>

When browsing printed output in the page browse mode, you can go from page to page by pressing the program function keys for FORWARD, BACKWARD, TOP, and BOTTOM. TOP takes you to the first page of output for the currently displayed procedure. If you are already on the first page, the TOP program function key displays the first page for the previously executed procedure.

<table>
<thead>
<tr>
<th>Command</th>
<th>SAS Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>GENERAL LINEAR MODELS PROCEDURE</td>
<td>GENERAL LINEAR MODELS PROCEDURE</td>
</tr>
<tr>
<td>DE</td>
<td>SUM OF SQUARES</td>
</tr>
<tr>
<td>1</td>
<td>10.00000000</td>
</tr>
<tr>
<td>2</td>
<td>0.00000000</td>
</tr>
<tr>
<td>3</td>
<td>0.00000000</td>
</tr>
</tbody>
</table>

Back on the program editor window, include your TEST file to show the CHANGE command. CHANGE A B changes the first occurrence of A to B. CHANGE * does the REPEAT CHANGE function. Also you can change every occurrence of B to A by entering C B A ALL.

In page browse mode, you can get a command line at any time by using program function key 4, and then you can execute line browse mode commands such as FIND.
The CAPS OFF command is useful for distinguishing between upper- and lowercase letters.

Finally, the NOFS command takes you out of the display manager and puts you into SAS line mode. It is the same as OPTION NOSOURCE NOFS command except that it also causes the screen to be cleared and display manager memory to be freed.
Pressing program function key 1 is just like typing HELP on the command line and pressing ENTER. To save time, you could type HELP ANOVA to get help on the ANOVA procedure, or you could simply type CHART and press program function key 1 to receive information on PROC CHART.

To resume the session in full-screen mode, enter OPTION FS;

To complete your introduction to the ability of the display manager to save you time, examine the program function key command equivalents, which are listed below.
Similarly, program function key 2 is equivalent to PRINT, and program function key 3 means END on the output window. Program function key 4 means RECALL on the program window, but it means command line toggle on the output window. Program function key 5 is for FIND; program function key 6 means CHANGE; program function key 7 is BACKWARD; program function key 8 scrolls FORWARD; program function key 9 is the split key; program function key 10 is LEFT; program function key 11 is RIGHT; and program function key 12 moves the cursor to the command line. In addition, the ENTER key means SUBMIT when there are no other commands pending on the program editor window.

All of the program function keys, including ENTER, are attention keys that cause the display manager to gain control. There are also some screen editing keys you may find useful. The home key puts the cursor on the first unprotected field of the screen; refresh redisplays the screen; the line feed key erases to the end of the field; and the backspace key toggles insert mode.

To exit from the system, submit an ENDSAS statement to the SAS compiler or enter BYE on any command line.