Developing a User Interface to Customize SAS* Output

Ty Lee, Eli Lilly and Company

Abstract

SAS* is well recognized as a development tool for information systems. It is not uncommon that a system should have a flexible interface to not only facilitate the request for output but also remotely control the output device without changing the default setup.

SAS/AF* and SCL are extremely helpful in building user interface. Based on author's experience on DREAMS (Data REporting And Management System) developed using SAS* on OS/2** with HP LaserJet** III as peripheral output device, this paper provides readers with the techniques to develop such a flexible output interface.

Following the procedure step by step, one can actually build a framework for a fairly sophisticated output interface. Due to the hardware independence since version 6.06, the method described can be adapted for other platforms as well.

Background

In one research area, statisticians and scientists work independently with their own SAS* programs and have to memorize program names with all the parameters. A re-engineering process has identified the benefits of consolidating all these end-user programs into a information system, through which the users can have shared access to data, software and hardware. The DREAMS (Data REporting And Management System) project is launched.

DREAMS information system is built on IBM PS/2 Model 90, using OS/2** 2.0 operating system with HP LaserJet** III as output device. SAS* 6.06 is selected as the development tool for the system. This includes but is not limited to SAS/SCL, SAS/AF* and base SAS*. etc. The critical part concerning the system output is as follows:

The critical part concerning the system output is as follows:

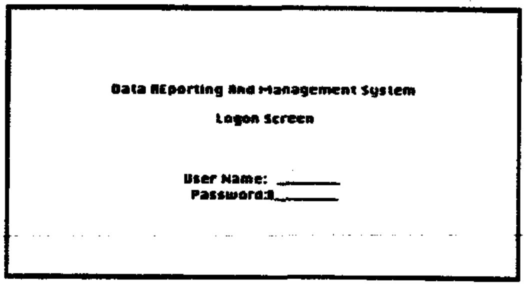
> ...
> There would be a user interface for output where user can choose whether he/she would like to browse the output on the screen or print it directly on the printer. Even after the user selects BROWSE, he/she should have the capability to print it out afterwards with specified format. Different reports generated on the printer may require different fonts, sizes, spacing, etc. The switch from one format to another should be transparent to users, and the default setup on the output device should remain the same all the time.

Procedures to build such a user interface will be explored in the following sections.

GUI under SAS/AF*

By using SAS/AF*, a logon screen can be developed which opens the system to only the authorized users. Once a user successfully logs on to the system, a pull-down menu would pop up on top of the window.

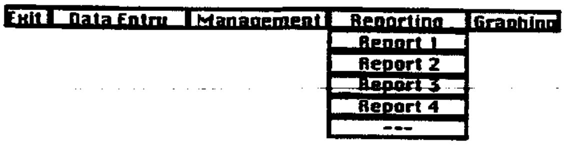
The PMENU facility is a menuing system that replaces the command line as a way to execute commands. It is available under SAS/AF* windows, SAS/FSP* windows, windows created with WINDOW or %WINDOW and the PROC REPORT window.



Data REporting And Management System

Logon Screen

User Name: _______
Password: _______

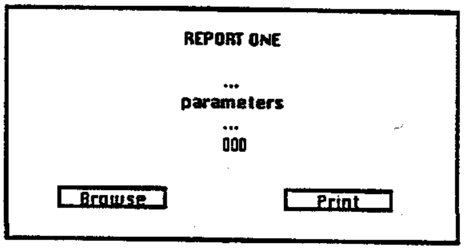


PMENU provides users with point-and-click functionality and presents a consistent interface to all parts of a system. And it eliminates the need for memorizing (e.g. program names, function keys, etc.).

The procedure to establish a PMENU is to, first of all, use PROC PMENU to create a PMENU entry that defines the desired action bar and pull-down menu hierarchies, and then associate PMENU catalog entry with the application window, finally, active the PMENU facility.

Browse or print - route of SAS* output

First of all, a window can be built for each output report by using SAS/AF*. On each window, there are two push buttons, BROWSE and PRINT specifically.



REPORT ONE

...
parameters
...
000

The following attributes are specified in the ATTR window:

Browse: Type=PUSHBTNC, Choice Group=ACTION, Protect=INITIAL, List=Browse
Print: Type=PUSHBTNC, Choice Group=ACTION, Protect=INITIAL, List=Print

In interactive SAS*, any output procedure would route output to the output window by default. And PROC PRINTTO can be used to redirect output to an external file or a hardware device. Therefore the algorithm behind the scene for DREAMS is:

```
If BROWSE is selected, then do nothing, use the default
If PRINT is selected, re-direct the output to a printer
```

Define and use forms in interactive SAS*

Output sent to OUTPUT window can also be printed. The PRINT command can print out everything in OUTPUT window. A better alternative is to go to OUTPUT

MANAGER window. A directory of OUTPUT files will display and a P can be placed to the left of an entry and that file will be printed. Multiple files can be selected for print at one time.

The selected files will be printed using the format pre-defined by SAS* default form, unless otherwise specified If different print formats are desired, a good approach is to define and use SAS* FORMs.

In interactive SAS*, FORM is used to specify the printer, text format, and other specifications. In this sense, how many different formats the user needs, the same number of FORMs will need to be created. Each would define one specific format and hopefully have a meaningful name.

Suppose a format is specified as 120 characters per line, 50 lines per page, portrait, You would start with a FORM by typing FSFORM *myform* at the SAS* command line to invoke the FORM window. The first frame pop up would be the Printer Selection. Since HP LaserJet** is used for DREAMS, You can click on Hewlett-Packard LaserJet** III [+]. Once you have made your printer selection, you cannot return to this frame.

Coming up next is the Text Body and Margin Information frame. If you are editing an existing FORM entry, the Text Body and Margin Information frame is the first frame you see when you invoke the FORM window. You can specify the following: 1) characters per line, 2) lines on first page, 3) lines on following, etc.

Your customized format would be defined in the Printer Control Language Frame by using escape sequences (reference the user's manual of your printer for details), also known as printer commands.

In your SAS* programs, use **OPTIONS FORMS=*myform*;** to utilize your customized form. All prints generated from display/output manager will be in the customized format defined in *myform*. You as a system developer can pre-define several forms (e.g. *myform1, myform2, myform3...*). For each specific report, say report1, use OPTIONS FORMS=MYFORM1 in the program REPORT1 to set the default form. User could browse the report and then issue a Print command to send it to the printer. After he issues a Print command in display/output manager, he would be prompted for a form name. If the user wants to use another format, at this time he could select any forms available by giving the form name.

Print directly

If user wants his output to be sent to printer directly, customized forms will be bypassed since prints will not be generated from display/output manager. In such a case, the way to use a specific format is to do the following:

a) Create a printer control language file: use any type of editor to enter the escape sequences of the printer. The following is an example of using Enhanced Editor in OS/2** for LaserJet** III: Type in Escape Sequences you need (Alt+27 would create the Esc character). Save this file as *myformat*

```
<-E
<-&a4L
<-&l0O
<-(8U
<-(s0p16.67h9v0s0b0T
<-&l6D
```

b) In your SAS* REPORTx.PROGRAM module, redirect all outputs to an external file instead of a printer by using **PROC PRINTTO PRINT=*outfile* new;**. At the end of the program, use the following commands to concatenate your printer control file with the outputs and send them to the designated printer:

```
FILENAME outfile "physical file name";
%LET control=physical file name;
%LET output=physical file name;

OPTIONS NOXWAIT;
        X "COPY &control +  &output  PRN";
OPTIONS XWAIT;
```

The output will be printed according to the format in &control file.
It's highly recommended that *myformat* will be named with font, spacing and other information presented so that others can make use of it also (e.g. PR8W122 - Portrait, font=Roman-8, 122 characters in width). It's also a good idea to come up with a directory of all the format files that are available in the system.

Remote-control the printer - the ultimate solution

Creating escape sequences files could be tedious. System developers have to hard code in all these printer control files according to user's format. Also the system described above is not flexible, since users cannot select or add alternative formats at run time.

In DREAMS, user can not only select BROWSE or PRINT when he/she requests a report, but also define a print format at the parameter screen, if he doesn't like the default format assigned to the report.

```
 %macro initprt;
/************************************************************/
/* Macro Name:   initprt                                    */
/* Purpose:    Redirect output to an external file for      */
 /*             formatted output.                           */
 /***********************************************************/
%global action;
 proc printto print=reports new;
 run;

 %let action= x "copy &control + &reports prn";
 run;
 %mend initprt;


 %macro doprint;
/************************************************************/
/* Macro Name:   doprint                                    */
/* Purpose:    Send output files to printer and then switch */
/*             back to default.                             */
/************************************************************/
proc printto;
 run;
```

```
%global action;
options noxwait;
        &action; /* use the OS/2** command to print */
options xwait;

data _null_;            /* reset the output file for future run */
  file reports notitle;
  put @20 "No output generated";
  put /@20 "Check SAS* log for details":
run;

title1 ' ';
footnote1 ' ';
%let action=;

%mend doprint;

%macro setform  (filename=,
                 pitch=10,
                 bold=1,
                 lmargin=0,
                 orient=0,
                 linespac=6);
/**************************************************************/
/*   Macro Name:  setform                                      */
/*   Purpose:  Create an escape sequences file which can       */
/*                    be used to customize text printout.      */
/*   Inputs:    &filename      - name of output test file,     */
/*                                  SAS* file name or physical */
/*                                  file in quotes             */
/*                    &pitch         - characters per inch, e.g. 10, */
/*                                  12, 16.67                  */
/*                    &bold  - 0 for normal, 1 for bold        */
/*                    &lmargin       - columns of left margin  */
/*                    &orient        - 0 for portrait,         */
/*                                  1 for landscape            */
/*                    &linespac      - lines per inch          */
/*   Outputs:  An escape sequences file                        */
/*   Reference:        Doris Wong, ARC                         */
/**************************************************************/
%if &bold=1 %then %let weight=3;
        %else %let weight=0;

data _null_;
  length esc $ 1;
  esc='1B'x;
  file &filename mod;
/**************************************************************/
/* Escape sequences:                                           */
/* esc E        = reset print format                           */
/* &a#L         = # columns for left margin                    */
/* &l#O         = # 0-portrait, 1-landscape, 3-reverse portrait, */
/*                   4-reverse landscape                       */
/* (8U          = Roman 8                                      */
/* (s0p         = fixed primary spacing                        */
/* (s#h         = # characters per inch                        */
/*  (s9v         = 9 points in height                                   */
/*  (s0s         = upright style                                        */
/*  (s#b         = # stroke weight                                      */
/*  (s3T         = courier                                              */
/*  &l#D         = # lines per inch                                     */
/***********************************************************************/
         put esc $1. 'E';
         put esc $1. '&a' "&lmargin.L";
         put esc $1. '&l' "&orient..O";
         put esc $1. '(8U';

         put esc $1. '(s0p' "&pitch.h" '0s' "&weight.b" '3T';
         put esc $1. '&l' "&linespc.D";
  run;
%mend setform;
```

Please notice that HP escape sequences are case sensitive and sometimes it's hard to tell whether it's 1 (one) or l (lower case of L). Also in escape sequences, "&" is the same used in SAS* macro variable name. This may cause some confusion too. The above technique is tested to be compatible on HP DeskJet** 500 and HP LaserJet** III. Make relevant modifications if a different printer is used.

Add the following in Main program or AUTOEXEC.SAS:

```
%let control=physical file name;
%let outputs=physical file name;
```

The program for each report will be as follows:

REPORTx.PROGRAM

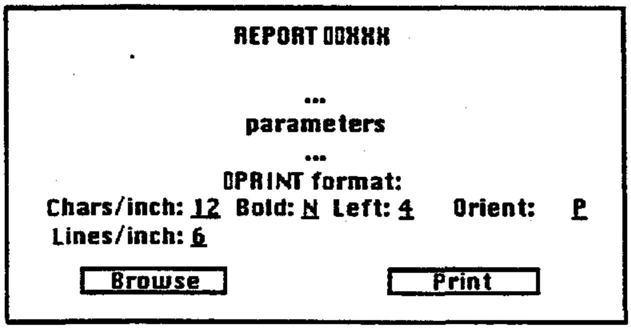
```
main:
  ...
  if modified(ACTION) then do;
     select(ACTION);
    when('BROWSE') do;
      call execcmd('END');
      end;
    when('PRINT') do;
            submit continue;  /* direct output to printer */
           %initprt
              run;
            endsubmit;
            call execcmd('END');
    end;
     end;
  end;
return;

term:
/* some AF* functions to validate SCL variables */
        ...
  submit continue;
        /* create a format */
        %setform(filename=&control, pitch=&P, bold=&B,
lmargin=&M, orient=&O, linespac=&L)
        run;
```

```
             /* SAS* program that generate report xxx */
             ...

             /* print and then direct output back to default */
             %doprint
             run;
     return;
```

Now the parameter screen for a specific report would look like this:



REPORT 00XXX

...

parameters

...

OPRINT format:

Chars/inch: 12 Bold: N Left: 4 Orient: P

Lines/inch: 6

In most cases, since default on the parameter screen is pre-defined by users, they would just use that. All they need to do is to click on the PRINT button to activate the procedure. In case that he/she desires a different format, he could basically update the format on this parameter screen and the printout will be customized dynamically.

DREAMS come true!

* SAS, SAS/AF and SAS/FSP are trademarks or registered trademarks of SAS Institute Inc. in the USA and other countries.

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

Ty Lee, Systems Analyst
Medical Information Systems
Drop code 2235
Eli LIlly and Company
Indianapolis, IN 46285
Phone: (317) 277-5183
Fax: (317) 277-3521
lee_ty@lilly.com