

CREATING INTERACTIVE USER WINDOWS WITHOUT USING SAS/AF™

Kathy Shelley, Iowa State University

Introduction

Computer hardware, software and high-speed communication lines enable us to obtain large amounts of data in very little time. To be able to quickly digest this information, it is most desirable to provide end-users with a menu interface which is easy to use. The WINDOW statement in SAS/Base™ provides this capability with little extra learning time for developers who normally program in the SAS language. The WINDOW facility is character-based and will run on any SAS system which uses the SAS Display Manager. Concise documentation with examples can be found in the *PC SAS Language Guide*.

WINDOW Statement Features

The WINDOW syntax is very similar to the use of PUT statements in SAS/Base. Color, window sizing and positioning can be specified. When users answer questions or specify choices, their responses are saved as values of variables in SAS data sets. Attributes such as reverse video or underline can be assigned to these fields in the windows. The values from these variables can then be used as decision criteria for running specified reports or options in other SAS programs. There is also a comparable %WINDOW statement in SAS' Macro language which will accept and pass macro variable values.

Applications Using the WINDOW Statement

Three different windowing applications will be presented in this paper. The first and most common is a window which presents a menu for selecting reports or graphs. The second guides the user through backing up data and programs from their PC hard disk to floppy diskettes. The third is an elementary approach to dynamic graphics where the user has the opportunity to interactively alter the appearance of a bar chart. This last application also demonstrates how to repeatedly display the graph through the use of the %DO macro statement.

Creating a Windowed Report Menu

Menus which allow the user to select one or more

reports/graphs comprise one of the most common uses of user input windows. This is very easy to do with the WINDOW statement. Following is a SAS window which allows up to 3 reports to be run. The SAS program for this window, saved as MENUTRAK.SAS, is given after the window.

```
• TRACKIT _____
• Command ==>
•
• MENU FOR TRACKING PROJECT VINCENT SAS USAGE
• Please choose one or more of the following reports by placing an X
• in the highlighted box(es):
•
•   ___ SAS Usage by Userid - Average Daily Usage and Total Users by Date
•
•   ___ SAS Usage by Workstation ID
•
•   ___ SAS Usage by Location (Building Name/Room Number)
•
• Enter PARTYON on the Command Line When Selections Are Complete
• Press <Ctrl> <Break> <Enter> to Stop Processing
```

- this is MENUTRAK.SAS;
- this program provides a windowed menu for choosing reports;
- which track the usage of sas on dec ultrix workstations;
- by userid, workstation id, or location (building name);

```
data a;
userid=' '; workstat=' '; freqloc=' ';

window trackit color=blue
#2 @5 'MENU FOR TRACKING PROJECT VINCENT SAS USAGE'
    c=yellow
#4 @5 'Please choose one or more of the following reports by placing an X'
    c=orange
#5 @5 'in the highlighted box(es):'
    c=orange
#9 @5 userid $char1. attr=rev_video autoskip=yes c=red
    +4 'SAS Usage by Userid - Average Daily Usage and Total Users by Date'
    c=cyan
#11 @5 workstat $char1. attr=rev_video autoskip=yes c=red
    +4 'SAS Usage by Workstation ID'
    c=cyan
#13 @5 freqloc $char1. attr=rev_video autoskip=yes c=red
    +4 'SAS Usage by Location (Building Name/Room Number)'
    c=cyan
#17 @10 'Enter' c=orange ' PARTYON ' c=white
    'on the Command Line When Selections Are Complete'
    c=orange
#20 @10 'Press <Ctrl> <Break> <Enter> to Stop Processing' c=cyan ;
do while (upcase(_cmd_) ne 'PARTYON');
display trackit blank;
end;
output;
stop;
```

```

run;

proc print data=a;
title "List of Menu Choices";
run;

data _null_;
set a;
file 'rept1.sas';
put ' ';
file 'rept2.sas';
put ' ';
file 'rept3.sas';
put ' ';

if userid ne '' then do;
file 'rept1.sas';
put "%include 'userid.sas';" ;
end;

if workstat ne '' then do;
file 'rept2.sas';
put "%include 'workstat.sas';" ;
end;

if freqloc ne '' then do;
file 'rept3.sas';
put "%include 'graf.sas';" ;
end;

run;

%include 'rept1.sas';
%include 'rept2.sas';
%include 'rept3.sas';

```

Hard-Disk Backup with Overlapping User-Input Windows

The following window is the first window displayed for guiding a user through a hard-disk directory backup to a specified diskette drive. After the user specifies Drive A or B, a second partial window is displayed asking for permission to continue the backup process. This example demonstrates how to execute an MS DOS™ command based upon the value from the window input variable DRIVENUM. It also shows how to display multiple windows on one screen.

```

*INTRO
*Command ==>
*
*      TREE GRADING DATABASE BACKUP:
*
*  After entering tree grading data, it may be desirable
*  to back up the data which you have entered. To do this,
*  you will need to have some diskettes ready. The backup will be done
*  using the DOS Backup command.
*
*  If you are planning to use 3 1/2 high density diskettes, you
*  will need to have 1 diskettes for this backup.
*
*  If you are planning to use 5 1/4 high density diskettes, you
*  will need to have 1 diskettes for this backup.
*
*      Please Enter Your Backup Diskette Drive Letter:
*      Enter either an A for Drive A or a B for Drive B : _
*
*

```

After the user enters A or B for the selected backup diskette drive, the following window appears requesting the user to continue with the backup. Notice that a new window named "INTRO3" is superimposed on the first "INTRO" window. If the response is Y, SAS will execute the DOS backup command via SAS' X facility.

```

*INTRO3
*Command ==>
*
*      TREE GRADING DATABASE BACKUP:
*
*  After entering tree grading data, it may be desirable
*  to back up the data which you have entered. To do this,
*  you will need to have some diskettes ready. The backup will be done
*  using the DOS Backup command.
*
*  If you are planning to use 3 1/2 high density diskettes, you
*  will need to have 1 diskettes for this backup.
*
*  If you are planning to use 5 1/4 high density diskettes, you
*  will need to have 1 diskettes for this backup.
*
*INTRO3
*Command ==>
*
*  If you have enough floppy diskettes, you
*  may continue with the backup.
*  Do you wish to continue? (Y or N) _

```

- this is BACKUP.SAS ;
- this program is an introductory sas window to state how many
- diskettes will be needed for last year's grading
- data backup using the dos backup command;

```

x'dir c:>c:\jerry.dir';
%include "sumbytes.sas"; *computes DISK3HD and DISK5HD vars;
%global drive partyon;

```

```

data _null_;
set new1;
drivenum=" ";
window intro color=black
#1 @25 "TREE GRADING DATABASE BACKUP:" c=orange
#3 @5 "After entering tree grading data, it may be desirable" c=cyan
#4 @5 "to back up the data which you have entered. To do this,"
c=cyan
#5 @5 "you will need to have some diskettes ready. The backup will be
done"
c=cyan
#6 @5 "using the DOS Backup command."
c=cyan
#10 @5
"If you are planning to use 3 1/2 high density diskettes, you" c=yellow
#11 @5
"will need to have" c=yellow
+1 disk3hd 3.0 c=white protect=yes autoskip=yes
" diskettes for this backup."
c=yellow
#14 @5
"If you are planning to use 5 1/4 high density diskettes, you"
c=orange
#15 @5
"will need to have" c=orange
+1 disk5hd 3.0 c=white protect=yes autoskip=yes
" diskettes for this backup."
c=orange
#18 @13 "Please Enter Your Backup Diskette Drive Letter:" c=cyan
#19 @13 "Enter either an" c=cyan " A for Drive A" c=white
" or a" c=cyan " B for Drive B" c=white " : " c=cyan
+3 drivenum $1. a=rev_video autoskip=yes required=yes;

```

```

window intro3 rows=20 rows=5 color=black
/2 @8 "If you have enough floppy diskettes, you"
      c=cyan
/3 @8 "may continue with the backup." c=cyan
/4 @8 "Do you wish to continue? (Y or N)" c=cyan
      +3 partyon $1. a=rev_video autoskip=yes
      required=yes
;
if upcase(drivenum) ne "A" and upcase(drivenum) ne "B"
  then goto display1; else return;
display1:
drivenum=" ";
display intro blank bell;
if upcase(drivenum) ne "A" and upcase(drivenum) ne "B"
  then goto display1;
call symput('drive',drivenum);

```

```

display intro3 blank bell;
call symput('partyon',partyon); stop;
run;

```

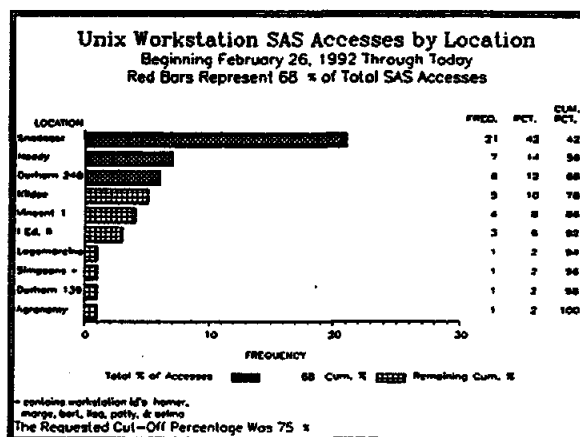
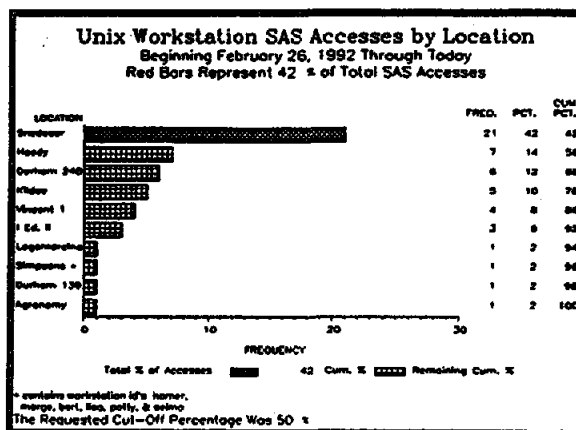
```

options symbolgen;
data a;
length partyon drive $1.;
partyon=trim(left(upcase("&partyon")));
drive=trim(left(upcase("&drive")));
if partyon="Y" and drive="A" then do;
  file 'drive.sas';
  put "x" backup "a" a;
end;
else if partyon="Y" and drive="B" then do;
  file 'drive.sas';
  put "x" backup "b" b;
end;
else if partyon="N" then do;
  file 'drive.sas';
  put 'run';
end;
options nosymbolgen;
run; %include 'drive.sas';run;

```

Dynamic Bar-Graphing

Dynamic coloring of bars can be achieved in PROC GCHART by redefining the *subgroup* variable values. The WINDOW statement, coupled with the %DO macro statement allows the user to repeatedly view the bar chart with different cut-off values for the coloring of the bars. When the user enters a cut-off value, it is passed to a Data step where a variable named CLASS is computed to be 1 for bar frequencies less than or equal to the cutoff value and CLASS is 2 for bar frequencies greater than the cutoff. The following two graphs result from the user specifying a 50% cut-off for the first graph and a 75% cut-off for the second:



The following window named PERCENT prompts the user for a cut-off percentage value for coloring the bars:

```

*PERCENT
*Command ==>
* INPUT WINDOW FOR CUMULATIVE PERCENTAGE CUT-OFF POINT FOR BAR CHART
*
* This window allows you to specify the cut-off percentage point
* for the bar chart which shows frequencies and percentages by
* location (building/room).
*
* If you would like to know which buildings total 75% of total SAS
* usage, you would enter 75 at the prompt below and the graph will
* color the bars red which cumulate up to 75% of total usage.
*
* Please enter a cut-off percentage: ___ and Press Return Key
* Note that the cut-off percentage must be greater than /this is computed from frequencies/
*
* *** To Produce the Graph, Enter PARTYON at the Command Prompt ***
* *** To Quit, Press <Ctrl> <Break> <Enter> ***
*
*_____P_____

```

The SAS program which produced the above window and the graphs follows on the next page.

```

* this is GRAF.SAS;
* datagraf.sas creates the output data set from proc freq named temp;
%include 'datagraf.sas';

```

```

%global encore;
data _null_;
  %let encore=Y;
run;

```

```

***** macro begins here *****;
** this is the window asking for the cutoff for cumulative percentages;

```

```

**** this gets the minimum cumpct value;
proc sort data=temp; by cumpct;
data new; set temp; by cumpct;
if _n.=1;
run;

```

```

***** this places the minimum cumpct value into macro var limit;
***** (the graph requires a cut-off >= this limit);
%global limit;
data _null_;
  set new;
  call symput('limit',cumpct);
**** macro again allows one to repeatedly display the graph
**** with different cut-off values ;

```

```

%macro again;
%do %while ( &encore=Y or &encore=y);
data a (keep=cutoff);
window percent color=black
  #3 @5 'INPUT WINDOW FOR CUMULATIVE PERCENTAGE CUT-OFF POINT FOR
BAR CHART'
  c=orange
  #5 @5 'This window allows you to specify the cut-off percentage point'
  c=cyan
  #6 @5 'for the bar chart which shows frequencies and percentages by'
  c=cyan
  #7 @5 'location (building/room). '
  c=cyan
  #9 @5 'If you would like to know which buildings total 75% of total SAS'
  c=cyan
  #10 @5 'usage, you would enter 75 at the prompt below and the graph will'
  c=cyan
  #11 @5 'color the bars red which cumulate up to 75% of total usage.'
  c=cyan
  #14 @5 'Please enter a cut-off percentage: ' c=white
  +1 cutoff $char3. attr=rev_video
  auto=yes required=yes c=red
  #14 @55 'and Press Return Key' c=orange
  #16 @5 'Note that the cut-off percentage must be greater than &limit'
  #19 @5 '*** To Produce the Graph, Enter ' c=orange
  'PARTYON ' c=yellow
  'at the Command Prompt *****' c=orange
  #20 @5 '*** To Stop Processing, Press <Ctrl> <Break> <Enter> *** '
c=cyan;

```

```

do while
  (upcase(_cmd_) ne 'PARTYON' or cutoff<&limit) ;
display percent blank;
end;
output a;
stop;
run;

```

```

footnote;
data temp; set temp; cutoff=' '; drop cutoff; run;
proc sort data=temp; by descending cumpct ; run;

```

```

data temp;
if _n.=1 then set a;
set temp;
run;
data temp;
set temp;
ycum=cutoff+0;
if ycum>=cumpct then class=1;
else if ycum<cumpct then class=2;
run;
proc sort data=temp; by class descending cumpct; run;
%global xclass yclass;
data _null_;
length xcum $3, yxcum $3;
set temp ;
if _n.=1;
x1cum=trim(cumpct);
xcum=trim(left(x1cum));
call symput('xclass',xcum);
y1cum=trim(ycum);
yxcum=trim(left(y1cum));
call symput('yclass',yxcum);
run;

```

```

proc sort data=temp; by location;
proc sort data=all; by location;
data both ;
  merge temp all end=eof; by location;
label class='Total % of Accesses';
run;

```

```

proc format;
value classfl
  1='&xclass Cum. %'
  2='Remaining Cum. %';
run;

```

```

***** proc gchart code follows *****;
pattern1 c=red v=s;
pattern2 c=blue v=m5x;
goptions cback=black colors=(white);
proc gchart data=both;
  axis1 length=60 pctl;
  hbar location / discrete
    subgroup=class
    descending
    freq pctl cpercent;
  title1 f=complex h=2.0 c=yellow
    'Unix Workstation SAS Accesses by Location';
  title2 f=simplex h=1.5 'Beginning February 26, 1992 Through Today';
  title3 f=simplex h=1.5 c=red
    'Red Bars Represent &xclass % of Total SAS Accesses';
  footnote1 f=simplex h=1 j=1 ' * contains workstation id's homer, ' ;
  footnote2 f=simplex h=1 j=1 ' marge, bart, lisa, patty, & selma';
  footnote3 f=simplex h=1.3 c=red j=1
    'The Requested Cut-Off Percentage Was &yclass %';
format class classfl.;
run;

```

```

data b;
xencore=' ';
window ancmore color=cyan
#3 @5 'Do you want to display the graph again with a new cutoff % value?'
  c=black
#5 @5 'Type Y or N : ' c=white
  +3 xencore $char1. attr=rev_video required=yes c=red
#8 @5 'Type ' c=blue ' STOP ' c=red
  'at the Command Line When Finished'
  c=blue;

```

```

do while (upcase(_cmd_) ne 'STOP');
display oncemore blank;
end;
output;
stop;
run;

```

```

data _null_;
set b;
call symput('encore',xencore);
run;
%end;
%mend again;
%again;
run;

```

This last window named, ONCEMORE, allows the user to either quit or call the first window and enter a new cut-off value. The SAS code for this window is on the previous page.

```

*ONCEMORE
*-----
*Command ===>
*
* Do you want to display the graph again with a new cutoff X value?
*
* Type Y or N:  _
*
* Type STOP at the Command Line When Finished
*-----P

```

Summary

In this paper, three different types of windowing applications have been presented along with the SAS code. The report menu and dynamic bar chart applications are currently being used in Ultrix SAS 6.07™ and PC SAS 6.04™. The backup example is running in PC SAS 6.04. The author wishes to thank Dan Parks for writing the SAS programs named USERID.SAS and WORKSTAT.SAS. His work provided the code for computing the frequency data set which was used in the bar graph examples.

Trademarks

™SAS, PC SAS, Ultrix SAS, SAS/AF, and SAS/Base are registered trademarks of SAS Institute. MS DOS is a product of Microsoft Corporation.

Author Information

The author can be reached as follows:

Kathy Shelley
e-mail: kathy@iastate.edu
phone: 515-294-7776
fax: 515-294-4040
117 Snedecor Hall
Department of Statistics
Iowa State
Ames, Iowa 50011