A SAS® Macro to Standardize Graphics
Swarna M. Ramanjulu, sanofi-aventis, Malvern, PA

ABSTRACT
Among the many buzz words in today’s FDA regulated pharmaceutical industry are “standards”, “compliance” and “reproducibility”. A lot of time is spent standardizing our tables and listings, but are your graphics standardized and presentable? Because graphs are like pieces of art, it is difficult not to have the programmer’s personal taste influence each graph. Thus, we end up with a variety of graphics. In an effort to standardize the appearance and streamline the production of graphics within our company, the decision was made to have one method of creating the graphical environment for all graphics via a SAS® macro.

A graphics macro was developed which incorporated uniform font, font size, a limited color palette for certain types of graphics, re-mapping of colors, and error checking. It also features the flexibility to produce outputs in various formats and is user-friendly. This paper will outline the features of the graphics macro including examples of code and output.

INTRODUCTION
Many companies use graphs in addition to tables and listings to present data. However, graphics can vary among programmers, depending on the graphics options (goptions) used. This paper will describe an easy way to control the appearance of graphs through a SAS® macro. The purpose of this macro is three-fold: 1) to improve the quality of the presentation of graphics, 2) to automatically set up the options for graphs depending on the output format desired, and 3) to produce graphs using limited colors for certain types of graphics.

SCOPE AND FEATURES OF THE MACRO
To facilitate the production of graphics and to standardize their appearance, a macro was developed which would produce in-text, appendix and ad hoc graphics with uniform goptions. The macro is compliant with company standards for graphics and includes the following features:

1. Produce in-text and appendix color graphics with a limited color palette and a built-in re-map feature for incorrect colors.
2. Set up goptions for graphs to include Times New Roman font and 10pt font size.
3. Direct output to screen, file or local hardcopy devices.
4. Error checking code.

DETAILS OF THE MACRO
The macro has four main parameters, &device, &gtype, &outname and &ttype. The parameter &device directs the graphics output to a display device such as the monitor, a file or a hardcopy device. The parameter &gtype identifies the type of graph desired, with possible inputs being intext, appendix or adhoc. &gtype also determines what colors-list would be used when producing color graphics. &outname is the name of the output file and &ttype is the format of the output, i.e. CGM.

For each combination of &device and &gtype, the goptions are pre-set and optimized for the specific output format. For example, if a user selects &device=cgmof97l and &gtype=adhoc, then a CGM file is created with all the colors available to that device. However, if a user selects &device=cgmof97l and &gtype=intext, then a CGM file is created using the limited color palette. In either case, a standard high quality graphic will be produced. The following paragraphs will demonstrate the features of the graphics macro and how results are achieved in more detail.

FEATURE 1: SET USER-DEFINED DEVICES AND MAP COLORS
Create user-defined devices with limited color palette using the gdevice procedure to copy and modify an existing device entry. Our standard output format for in-text and appendix graphics is CGM; therefore, the cgmof97l driver was modified. The device parameter, Maxcolors, limits the colors for the modified device, and the parameter, colors, specifies the colors to be used in the output. Any color that is not in the colors-list will be automatically re-mapped to the first un-used color in the colors-list. Thus, the first feature of the macro is achieved.

```
proc gdevice catalog=work.devices nofs;
   copy CGMCEF97L from = sashelp.devices newname = CGMCEF97C;
   modify CGMCEF97C;
   description='CGMCEF97L with new colors list';
   Maxcolors = 7;
   Colors=(CX000000 WHITE CXFF0000 CX008000 CX996633 CXFF00FF CX00FF00CX);
quit;
```
Create the colors with specific hexadecimal code as global macros and include them in the colors-list.

```
%global red black green brown purple orange;
%let red=CXFF0000;
%let black=CX000000;
%let green=CX008000;
%let brown=CX996633;
%let purple=CXFF00FF;
%let orange=CXFF6600;
```

%if %lowcase(&gtype)=appendix or %lowcase(&gtype)=intext
  %then %do;
    %let xcolor=(black red green brown purple orange);
  %end;
%else %if %lowcase(&gtype)=adhoc
  %then %do;
    %let xcolor=( );
  %end;

FEATURE 2: MAP GOPTIONS TO OUTPUT REQUIREMENTS

Set up the macro for the desired graphic output with the goptions pre-set for each device driver. In the example of the partial code shown below, if &gtype=intext or appendix, then the modified device is selected and the appropriate colors-list in the goptions set. Note that the goptions are reset for each driver. In this way a standard graphic will be produced with uniform font, font size, colors and size of graphic, regardless of output type.

```
%if (&device=cgmof97l) and %upcase(&rotate)=LANDSCAPE
  %then %do;
    %if (&gtype=intext or &gtype=appendix) %then %do;
      %let dev=CGMOF97C;
      %let mode=replace;
      %let ftext=HWCGM005;
      filename gsasfile "&outlib./&outname..&ttype";
      goptions reset=all;
      goptions device=&dev gaccess=gsasfile gsfmode=&mode ftext=&ftext colors=&xcolor
      hpos=80 vpos=50 htext=1.2 htitle=1.2 rotate=landscape cback=white lfactor=2;
    %end;
  %else %if &gtype=adhoc %then %do;
    %let dev=CGMOF97L;
    %let mode=replace;
    %let ftext=HWCGM005;
    filename gsasfile "&outlib./&outname..&ttype";
    goptions reset=all;
    goptions device=&dev gaccess=gsasfile gsfmode=&mode ftext=&ftext colors=&xcolor
    hpos=80 vpos=50 htext=1.2 htitle=1.2 rotate=landscape cback=white lfactor=2;
  %end;
%end;
```

FEATURE 3: DIRECTING OUTPUT

The third feature of the macro gives the user the flexibility to send output to a display device such as a monitor, to local hardcopy devices or to a file. In order to preserve the same limited color palette for in-text and appendix graphics, the color printer device was modified similar to the code in Feature 1. As a result, the hardcopy resembles the actual graphic in terms of the colors used. Ad hoc color graphics would use all colors available to the local hardcopy device.

```
proc gdevice catalog=work.devices nofs;
  copy PHASR540 from = sashelp.devices newname = PHASR740;
  modify PHASR740;
  description='NEW PHASR740 - PHASR540 color printer with new colors list’;
  Maxcolors = 7;
  Colors=(CX000000 WHITE CXFF0000 CX008000 CX996633 CXFF00FF CXFF6600);
quit;
```

In a similar manner, the device WIN was modified to send output to the screen during development. The modified device, WINX, in this case, would display the re-mapped colors as the actual colors of the graph.

```
%if &device=screen %then %do;
  %if (&gtype=intext or &gtype=appendix) %then %do;
    %let dev=WINX;
  %end;
%end;
```
FEATURE 4: ERROR CHECKING
The last feature of the macro is the built-in error checking code. Error checking facilitates debugging and allows for a more user-friendly macro. Here is an example of how this is set up in the macro:

```sas
%if %lowcase(&gtype) ne appendix and %lowcase(&gtype) ne intext and %lowcase(&gtype) ne adhoc %then %do;
    %let dev=WIN;
    goptions reset=all;
    %let ftext="Times New Roman";
    goptions device=&dev targetdevice=cgmof97l ftext=&ftext colors=&xcolor hpos=80 vpos=50 lfactor=2 rotate=landscape cback=white;
    %end;
%else %if &gtype=adhoc %then %do;
    %let dev=WIN;
    goptions reset=all;
    %let ftext="Times New Roman";
    goptions device=&dev targetdevice=cgmof97l ftext=&ftext colors=&xcolor hpos=80 vpos=50 lfactor=2 rotate=landscape cback=white;
    %end;
%end;
```

EXAMPLES
The graphics macro is invoked by passing the required parameters. This will produce graphics in your output directory. In the examples shown below, the program which creates the graphs uses blue for Treatment A, green for Treatment B and brown for Treatment C. The code for producing both graphs is identical, except for the change in values for &gtype.

FIGURE 1: CGM PRODUCED WITH &GTYPE=ADHOC

Note: Blue is NOT a company standard color; However, since &gtype=adhoc, blue appears as it should in the output. The cgmof971 driver is used with all the colors available to the device.
CONCLUSIONS
The graphics macro is the one method used to produce a standardized graphical environment for all graphics within our company. It was designed to create output conforming to company standards regarding the appearance of graphs, including the use of color for certain types of graphs. Additional enhancements and features could be added to the macro to support different companies’ standards or output desired. SAS/GRAPH may have limits, but it is still a very powerful tool and can be easily used to create high quality graphs. By producing all graphics through a set-up macro like %graphics, much of the preliminary work is completed leaving only the production of the graph itself.

REFERENCES

ACKNOWLEDGMENTS
Many thanks to my boss, Greg Ridge, for “volunteering” me for the Graphics project. Much appreciation to my friends and colleagues for the many hours spent proofreading this paper. Finally, thanks to my husband and my dad who continue to be my inspiration.

SAS and SAS/GRAPH is a Registered Trademark of the SAS Institute, Inc. of Cary, North Carolina.

CONTACT INFORMATION
Your comments and questions are valued and encouraged. Contact the author at:

   Swarna M. Ramanjulu
   sanofi-aventis
   9 Great Valley Parkway
   Malvern, PA 19355
   Work Phone: 610 889-8867
   Fax: 610 889-6932
   Email: swarna.ramanjulu@sanofi-aventis.com

* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

Note: Treatment A, originally in blue, is re-mapped to &red. Blue is not a standard color for in-text graphics.