Paper 062-31

A Powerful Macro to Control Title Appearance in SAS/GRAPH® Output
Louie Huang, Baxter BioScience, Westlake Village, California 91362, USA

ABSTRACT

Whether or not to have titles in the graphs is always out of your control in clinical trials. It is a painful job when you are asked to add titles into hundreds of graphs at the last minute of your project deadline. The more graphs you have, the more time you need to spend. A macro that can control title appearance is absolutely valuable when creating multiple graphs is necessary.

This paper introduces an efficient and straightforward macro used to automatically handle the title appearance of SAS/GRAPH® output in CGM (Computer Graphics Metafile) format created from a varying number of programming files. Using this macro, you can execute hundreds of separate programming files to create graphs with titles or figure captions, or without titles. The paper discusses problems by practical examples, and summarizes function of the macro and the related programming skills. The macro is conveniently adaptable to generate multiple formats of SAS/GRAPH® output simultaneously. After reading this paper, you should be able to easily deal with the problems in your own work.

INTRODUCTION

When generating graphs from a varying number of programming files, you have to consider the unique requirements and specifications for a given type of graphs. One of the most frustrating and painful aspects is that you are never certain whether titles are needed or not, or how many lines of titles will be for a given graph. In my past programming practice of clinical trials, I was often asked to generate CGM graphs without titles, on which I already spent a lot of time to create the programming files with title statements. Then, I had to walk through the programming files to block the title statements and modify some graph options in order to make the graphs fit perfectly. Sometimes, I was told to make hundreds of CGM graphs without titles throughout the study period. However, I was asked to add titles into my well-developed graphs at the last minute of the project deadline. This kind of work often becomes tedious and time-consuming, because adding titles is much more difficult than blocking the title statements. A powerful macro program is developed in order to efficiently handle with the uncertain programming tasks discussed above.

PRACTICAL PROBELEMS AFFECTING GRAPH OUTPUT

I would like to introduce a practical example, in order to help you realize the problems in your real work. Figure 1 (A) was originally created with no titles by setting length=78 and 80 for horizontal and vertical axis, respectively. Figure 1 (A) looks good. Figure 1 (B) was modified from Figure 1 (A) by adding the titles into the graph. When you looked into the LOG window, a warning message said "WARNING: The left vertical axis labeled *wtpre* was too large as specified. Specify LENGTH=71.3". You probably think it is not a big deal without having to check the graph output. Actually, there are two problems in Figure 1 (B): 1) the titles are

overlapped with the graph frame, and 2) the horizontal label "Drug" is partially cut off. To solve the problems, you need to leave enough space for the titles by reducing the length of the vertical axis. You might think the simple solution is to remove the LENGTH options from the axis statement of your programming code and let the SAS$^{®}$ figure it out. Unfortunately, it doesn't work out that way! Figure1 (C) was created by omitting the LENGTH option. The right frame of Figure 1 (C) was cut off. Obviously, Figure 1 (D) is what you want. It was generated by setting **_LENGTH=80_** and **_LENGTH=67_** for the horizontal and vertical axis, respectively.

After a detailed discussion, I would like to emphasize the function of the LENGTH option. It controls the dimension of horizontal or vertical axis by the units specified in the GUNIT= option in GOPTIONS statement and the number specified in the LENGTH= option in the axis statements. When you want to introduce titles into your graphs, you have to leave enough room for the titles by assigning a small number to the length of the vertical axis.

Furthermore, I would like to mention several other SAS/GRAPH$^{®}$ options that affect the position of your graph output. HSIZE= and VSIZE= determine the external dimensions of your graph. HORIGIN= and VORIGIN= set the horizontal and vertical offset, respectively, from the lower-left corner of the display area to the lower-left corner of the graph. Please be aware that: 1) PDF output is required to set both HORIGIN= and VORIGIN=; 2) not necessary to set HORIGIN= and VORIGIN= for CGM output; 3) not necessary to set HORIGIN= for RTF output.

It is not surprising at all for an inexperienced programmer to take hours to optimize the code of one typical graph by trial and error. Also, you should be aware that adding titles into graphs one by one is not only very boring but also time-consuming. Furthermore, whether or not to have titles in the graphs is always out of a programmer's control. Therefore, the author intends to introduce an efficient yet powerful macro to control the title appearance. You are able to tremendously reduce your workload by using this macro.

The rationale of this method is summarized as follows: 1) the text of your graph titles is managed in Excel$^{®}$; 2) a SAS data file containing the title information is loaded into SAS$^{®}$ with an IMPORT procedure; 3) The titles are converted into lists of macro variables by PROC SQL, in which a character '#' instructs SQL to sequentially append the titles from the SAS data file; 4) the macro **_ctrl_tit_** is conditionally executed at your client's preference i.e. titles or not, or captions only. Once you learn this macro, no matter whatever decision is made for you, your life is always easy.
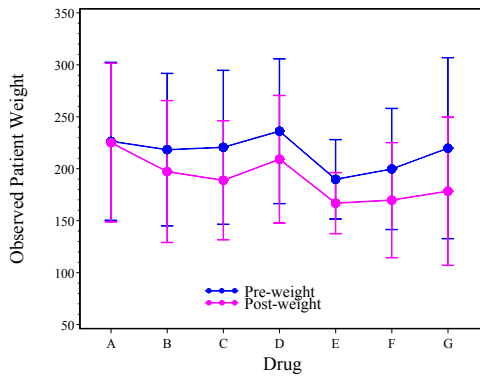
Figure 1. Graph Example

ANNOTATION AND INTERPRETATION OF THE MACRO
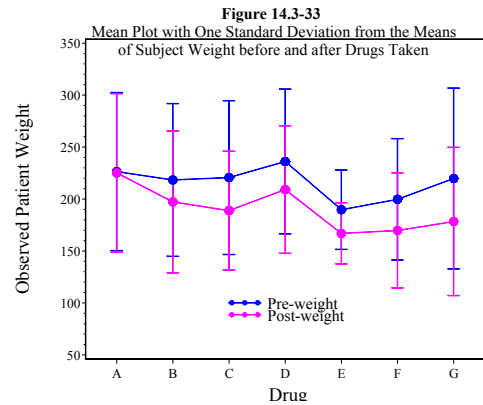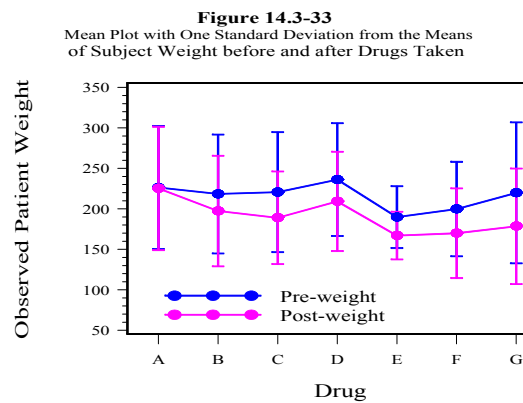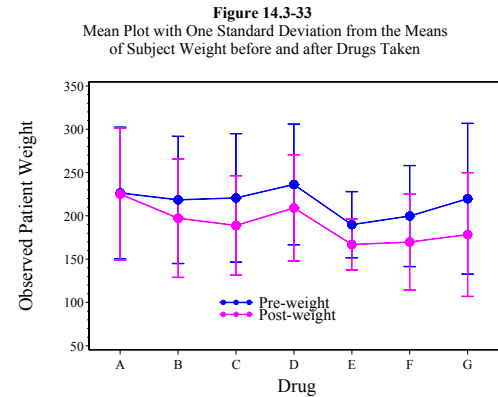
Macro *ctrl_tit* is displayed in Figure 2. The key portion of the macro is numbered at the left for the convenience of interpretation.

The text of all titles is entered into an EXCEL® spreadsheet. The text between each title is separated by a pound (#) sign. If some graphs lack the 4th title, the text is replaced with XXX.

(1) An IMPORT procedure in the SAS file *PROG_DATA.SAS* converts the text file into a SAS data file. This SAS file also contains the manipulated data sets and graph format information.

(2) A SQL procedure converts the text of title1, title2, title3, and title4 into macro variables *&list1*, *&list2*, *&list3*, and *&list4*, respectively.

(3) The value of the automatic macro variable *&sqlobs* is equivalent to the number of graphs to be created and the number of SAS files to be executed.

(4) A macro variable *dir* is placed into the FILENAME statement of each SAS file, which is used to generate a graph. The title statement is inactivated by placing a star(*) before it, when needed. Therefore, this macro can be conditionally executed. If titles are required, the code from line 5 to 14 brings the titles into your graphs, and exports the graphs into the *title_yes* directory. If titles are not required, the code from line 16 to 24 blocks the title statements, and exports the graphs into the *title_no* directory. If only figure captions are required, the code from line 26 to 34 activates the first title statement and blocks the other title statements, and exports the graphs into the *title_cap* directory.

(5) The code in lines 10 and 11 is valid for the graphs that require a 4th title.

(6) The macro variable *&titnum* determines the length of vertical axis. The value of macro variable *&titnum* is equivalent to the number of titles. Each title takes about 4 percent of the vertical length. The vertical length of the graph is determined by the following formula: **vlen=%sysevalf((80-3.75*&titnum),floor) .**

(7) If a title of any graph is "too long", the CALL SOUND function will make noise to remind of you. You can check the LOG window to identify which titles of which graphs are too long.

(8) A %INCLUDE statement reads the programming files into SAS® to create graphs.

Figure 2. Core Coding of the Macro

```
goptions reset=global gunit=pct hsize=7 in vsize=6.5 in hsize=7 in vsize=6.5 in
         colors=(black green pink red yellow magenta blue) cback=white ctext=black
         ftitle="TimesRomanBold"  ftext="TimesRoman" htitle=3 htext=2.5;
%include "C:\SUGI31\title_controller\prog_data.sas"; (1)
proc sql noprint; (2)
 select title1,title2,title3,title4
        into :list1 separated by '#',:list2 separated by '#',
             :list3 separated by '#',:list4 separated by '#'
        from tit.title_text; quit;
 1   %macro ctrl_tit(tit,titnum,nbyes);
 2     %global t1 t2 t3 t4 hlen vlen;
 3      %do i=1 %to &sqlobs; (3)
 4   /* place full titles into graphs */
 5   %if %upcase(&tit)=YES %then %do;
 6    %let dir=title_yes; (4)
 7   %let t1=title;  (4) /* valid title statements */
 8   %let t2=title2;
 9   %let t3=title3;
10   %if %scan(&list4,&i,#)=XXX %then %let t4=*title4; /* block the 4ᵗʰ title */
11   %else %let t4=title4; (5)                 /* activate the 4ᵗʰ title */
12   %let hlen=80;
13   %let vlen=%sysevalf((80-3.75*&titnum),floor); (6)
14   %end;
15   /* No tiltes  */
16   %else %if %upcase(&tit)=NO %then %do;
17   %let dir=title_no; (4)
18   %let t1=*title; (4)    /* block the title statements */
19   %let t2=*title2;
20   %let t3=*title3;
21   %let t4=*title4;
22   %let hlen=80;
23   %let vlen=%sysevalf((80-3.75*&titnum),floor); (6)
24   %end;
25   /* Figure cations only */
26   %else %if %upcase(&tit)=CAP %then %do;
27   %let dir=title_cap; (4)
28   %let t1=title; (4)    /* only the title statement for figure caption is valid */
29   %let t2=*title2;
30   %let t3=*title3;
31   %let t4=*title4;
32   %let hlen=80;
33   %let vlen=%sysevalf((80-3.75*&titnum),floor);(6)
34   %end;
35   /* extract the titles from the macro vaiables generated by SQL */
36   %let tit1=%scan(&list1,&i,#);
37   %let tit2=%scan(&list2,&i,#);
38   %let tit3=%scan(&list3,&i,#);
39   %let tit4=%scan(&list4,&i,#);
40   data _null_;  /* A noise will sound if title is too long when excuting */
41   if %length(&tit2) ge &nbyes or %length(&tit3) ge &nbyes or %length(&tit4) ge
      &nbyes
42    then call sound(600,2400); (7)
43   run;
44   %if %length(&tit2) ge &nbyes %then %put &tit1 &t2 is too long; (7)
45   %if %length(&tit3) ge &nbyes %then %put &tit1 &t3 is too long;
46   %if %length(&tit4) ge &nbyes %then %put &tit1 &t4 is too long;
47   %include "C:\SUGI31\title_controller\&tit1..sas"; (8)
48   %end;
49   %mend;
```

Appendix I presents a modified version of the macro *ctrl_tit* that has more versatile functions. The parameters of the macro variables are interpreted as follows: ***&tit*** - contents of titles; ***&nbyes*** - length of the title; ***&ftit*** - title font; ***&ftx*** - text font; ***&dv*** – device option; ***&exd*** – graph format option; ***&hor*** - option for horizontal origin; and ***&vor*** – option for vertical origin. When you execute the modified version, you are able to simultaneously export multiple formats of graphs with or without titles, or with figure captions only to multiple locations you specified in the macro.

CONCLUSION

Adding and removing titles not only offers a lot of tedious hand manipulation, but also becomes error-prone. This paper presents a versatile macro that has great flexibility to make your programming life a lot easier when you need to create hundreds of graphs from many separate SAS files.

CONTACT INFORMATION

The author appreciates any comments, questions, and suggestions for improvement.

Louie Huang
Baxter BioScience, Baxter Healthcare Corporation
One Baxter Way
Westlake Village, CA 91362
Tel: 805-372-3487
Fax: 805-372-3462
Email: louie_huang@baxter.com

Disclaimer

APPENDIX I – A MACRO CODE WITH VERSATILE FUNCTIONS

```
%include "C:\sugi31\title_controller\multiple_device\prog_data.sas";
options mlogic mprint;

/* titles are converted into clustered macro vaiables separated by a separator # */
proc sql noprint;
 select title1,title2,title3,title4
        into :list1 separated by '#',:list2 separated by '#',
             :list3 separated by '#',:list4 separated by '#'
        from tit.title_text
quit;

%macro ctrl_tit(tit,nbyes,ftit,ftx,dv,exd,hor,vor);
 %global t1 t2 t3 t4 hlen vlen;

 goptions reset=global gunit=pct
         hsize=7 in vsize=6.5 in  horigin=&hor in vorigin=&vor in
         colors=(black green pink red yellow magenta blue)
         cback=white ctext=black ftitle=&ftit
         ftext=&ftx htitle=3 htext=2.5;

 %do i=1 %to &sqlobs;
/* Bring titles into graphs */
%if %upcase(&tit)=YES %then %do;
 %let dir=title_yes;
 %let t1=title;    /* valid title statement */
 %let t2=title2;
 %let t3=title3;
 %if &i=4 %then %let t4=title4;
 %else %let t4=*title4;
 %let hlen=79;
 %let vlen=65;
%end;
/* No tiltes  */
%else %if %upcase(&tit)=NO %then %do;
 %let dir=title_no;
 %let t1=*title;     /* title statement is blocked by placing a * before it */
 %let t2=*title2;
 %let t3=*title3;
 %let t4=*title4;
 %let hlen=79;
 %let vlen=65;
%end;
/* Figure cations only */
%else %if %upcase(&tit)=CAP %then %do;
 %let dir=title_cap;
 %let t1=title;
 %let t2=*title2;
 %let t3=*title3;
 %let t4=*title4;
 %let hlen=79;
 %let vlen=65;
%end;
 /* extract titles from the macro vaiables generated by SQL */
 %let tit1=%scan(&list1,&i,#);
 %let tit2=%scan(&list2,&i,#);
 %let tit3=%scan(&list3,&i,#);
 %let tit4=%scan(&list4,&i,#);
```

```
    data _null_;   /* A noise will remind of you if your title is too long */
if %length(&tit2) ge &nbyes or %length(&tit3) ge &nbyes or %length(&tit4) ge &nbyes
    then call sound(600,2400);
run;
%if %length(&tit2) ge &nbyes %then %put &tit1 &t2 is too long;
%if %length(&tit3) ge &nbyes %then %put &tit1 &t3 is too long;
%if %length(&tit4) ge &nbyes %then %put &tit1 &t4 is too long;
filename grafout "C:\huang\temp\sugi31\title_controller\graph\&dir\&tit1..&exd";
goptions device=&dv gsfname=grafout gsfmode=replace;
%include "C:\sugi31\title_controller\multiple_device\&tit1..sas";
%end;
%mend;


/* creates CGM graphs with full titles in hardware fonts */
%ctrl_tit(yes,54,"TimesRomanBold","TimesRoman",cgmof97p,cgm,0,0)
 /* creates CGM graphs without titles */
%ctrl_tit(no,54,"TimesRomanBold","TimesRoman",cgmof97p,cgm,0,0)
/* creates CGM graphs with figure caption in hardware fonts */
%ctrl_tit(cap,54,"TimesRomanBold","TimesRoman",cgmof97p,cgm,0,0)
 /* creates PDF graphs with full titles in software fonts */
%ctrl_tit(yes,54,Swissb,Swiss,pdfc,pdf,0.75,2.25)
 /* creates PDF graphs without titles */
%ctrl_tit(no,54,Swissb,Swiss,pdfc,pdf,0.75,2.25)
/* creates PDF graphs with figure captions in software fonts */
%ctrl_tit(cap,54,Swissb,Swiss,pdfc,pdf,0.75,2.25)
quit;
```