PharmaSUG2011 – Paper SP02

Incorporating Graphics into Summary Report Tables using ODS and GTL

Qinghua (Kathy) Chen, Genentech Inc., South San Francisco, CA

ABSTRACT

Graphical representation of treatment effect is often easier to understand and therefore frequently used in conjunction with summary reports in Clinical Study Reports. The roll out of SAS® Output Delivery System (ODS) has facilitated the automatic creation of statistical graphics and ODS Graphics Template Language (GTL) when used with a data step or PROC SGRENDER procedure allows you to design your own layout so you can easily incorporate the statistical graphics into summary reports. The paper will describe how to use ODS Graphs and GTL to incorporate dynamic visualization of data within the body of a summary report table.

Topics include:

- Creating statistical graphs from SAS procedure with ODS
- Customizing summary reports by incorporating graphics using GTL

INTRODUCTION

Dynamic data visualizations can usually be interpreted much more quickly than numbers in summary tables and thereby prove to be extremely valuable within Clinical Study Reports. For certain types of analyses such as survival analysis, there is often two sets of output: a summary table of statistics and a separate graph. In an effort to improve readability, the concept of presenting a graph and a summary report together was thought to be an ideal solution although it was not readily achievable. For example, the creation of a forest-plot or “blobbogram” was only possible using an annotated data set in previous versions of SAS software. Unfortunately, while the end product was visually appealing, it was not very flexible in terms of different layouts and the learning curve was quite steep.

Beginning with version 9.2, SAS provides innovative, new ways of creating graphics. The ODS graphics language enables us to create high quality graphs and control the location, name of the output, graph dimensions and output types easily. GTL is a subset of the PROC TEMPLATE statements used with STATGRAPH templates which allows us to define our own graphics layout and style. Now incorporating a graph into a SAS summary report is much easier than it previously was using an annotated data set. Therefore it increases our programming efficiency.

CREATING STATISTICAL GRAPHS FROM SAS PROCEDURE WITH ODS GRAPHICS

There are four SAS/GRAPH statistical graphics procedures, SGPLOT, SGPANEL, SGSCATTER and SGREDER, and each of them has their own purpose. Unlike the first three, the SGRENDER procedure allows you to create graphs based on templates you create. Since it uses GTL, it is the most flexible procedure. The ODS graphics statement controls the physical aspects of the graph, such as image size, name of the output, etc., and it affects all the graphs that are rendered in a SAS session until overridden by another ODS graphics statement. The example below shows you how SGRENDER is used to execute a template and direct a .pdf output to the pdf directory with foreststyle style defined in proc template.

```
ods listing image_dpi=300 gpath='../pdf' style=foreststyle;
ods graphics / imagename="&jobname" imagefmt=pdf;
proc sgrender data=fortable template=ForestPlot;
run;
ods listing close;
ods graphics off;
```

IMAGEFMT specifies the format of the output. It can be HTML, PDF, RTF, PNG and GIF etc.

CUSTOMIZING SUMMARY REPORTS BY INCORPORATING GRAPHICS USING GTL

STATGRAPH and STYLE are two of a few templates that PROC TEMPLATE supports. The graph template defines the layout and detail of the output to be produced. It includes statements for graph layout (lattice or overlay,) plot type (scatter plots or histograms,) text elements (titles, footnotes and insets) etc. The style template defines the format information for the visual

SAS output. GTL is a compiling language, so it will accept runtime variable substitution via dynamics or macro variables. For STATGRAPH such variables should be declared within the template definition before the BEGINGRAPH block. Each of the DYNAMIC, MVAR, and NMVAR statements can define multiple variables and values for dynamics that resolve to column names or strings should be quoted. The current symbol table (local or global) will be used for macro variable values at runtime.

There are three basic steps to create a graph using GTL.

1. Use PROC TEMPLATE to define a STATGRAPH template
2. Define STYLE template
3. Create the graph by running the SGRENDER procedure with appropriate data

EXAMPLE 1: CREATING A FOREST-PLOT (BLOBBOGRAM) IN A SUMMARY REPORT

1. Use PROC TEMPLATE to define a STATGRAPH template

Define STATGRAPH ForestPlot for the structure of graph to be produced:

```
define statgraph ForestPlot ;
```

Define macro variables SI that will resolve to numbers for bubble size in the graph:

```
nmvar %do i=1 %to &nobs;
           s&i
      %end;;
```

Define the dimension of the graph:

```
begingraph / designwidth=1000px designheight=600px;
```

Define title:

```
entrytitle "Table 1" /  pad=(bottom=5px);
```

Define output layout: The following statement creates a grid of graph cells with 10 columns and each column width will be proportional to the values specified in columnweights. We can use external AXES for the graph with lattice layout.

```
layout lattice / columns=10   columngutter=0
                 columnweights=(.14 .04 .04 .05 .14  .04 .05 .14 .15 .21);
```

Define text column in the graph: The y2axisopts is used to define second yaxis in the graph for Baseline Risk Factors, so the values can be left justified in display. You may have to adjust the offsetmin and offsetmax in order to make the summary report well aligned to the graph on the right as SAS will align Y2 to Y axis by default, but not on the offset. In this example I have used 0.09 for both offsets. CONSTANT is a dummy variable set equal to one and is used to overlay text on the same graph.

```
       layout overlay/walldisplay=none border=false
                    y2axisopts=(reverse=true type=discrete display=(tickvalues)
                    offsetmin=0.09 offsetmax=0.09)
                    xaxisopts=(display=none offsetmin=0 offsetmax=0);
              entry halign=left "Baseline Risk Factor"/textattrs=GraphLabelText location=outside
valign=top;
              scatterplot y=her2 x=constant/yaxis=y2 markerattrs=(size=0);
       endlayout;
```

Define the last column in the graph:

```
layout overlay / walldisplay=(fill)
                 yaxisopts=(display=none reverse=true  offsetmin=0.1 offsetmax=0.1
                            linearopts=(integer=true))
                 xaxisopts=(type=log offsetmin=0 offsetmax=0 label="       "
                 logopts=(base=10 tickintervalstyle=logexpand
                          minorticks=true viewmin=0.01 viewmax=100 ));
```

AXES can be of different type, discrete, time, linear or log. From the example above, you can see I used linear for y axis, but use log for the x axis in order to have a balanced look in graph.

.

.

Draw the horizontal lines of 95% CI of Odds Ratio:

```
seriesplot y=keyvargp x=value / group=keyvargp lineattrs=(pattern=solid);
```

Draws the Odds Ratios (one for each observation) with a dynamic size applied to it at run time. The size value was obtained from macro variables created by PROC SQL. Since SAS does not allow us to use variable values for graphic attributes yet, the odds ratios must be drawn one-by-one with different bubble sizes as specified by the macro variable create above.

```
%do i=1 %to &nobs;
     scatterplot x=c&i y=v&i / markerattrs=(color=black symbol=circle size=s&i);
 %end;;
```

The following statement defines where a vertical reference line is drawn in the graph.

```
Referenceline x=0.1 / lineattrs=(pattern=shortdash);
```

Define footnote:

```
entryfootnote halign=left "CI = Confidence interval." /pad=(bottom=0px);
```

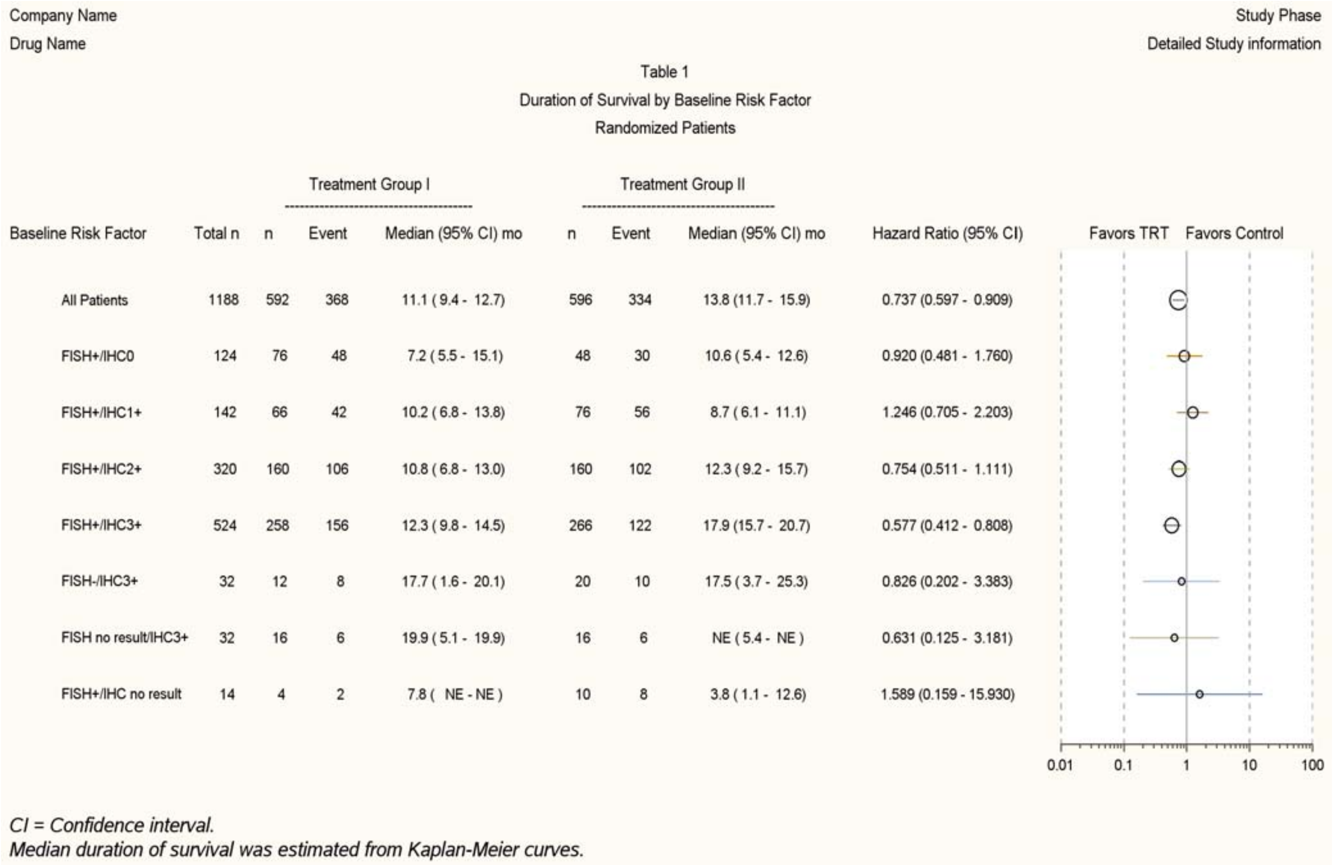
2. Define STYLE template

```
proc template;
      define Style foreststyle;
             parent = styles.analysis;
             style GraphFonts from GraphFonts
             "Fonts used in graph styles" /
             'GraphTitleFont' = (", ",8pt)
             'GraphLabelFont' = (", ",8pt)
             'GraphValueFont' = (", ",7pt)
             'GraphDataFont' = (", ",7pt);
       end;
run;
```

3. The SGRENDER procedure is bound to the ODS object at run time and directs the graphic output to the ODS destination.

```
proc sgrender data=fortable template=ForestPlot;
run;
```

Here is the output of example 1 (Fig. 1).



Company Name   Study Phase
Drug Name   Detailed Study information

Table 1
Duration of Survival by Baseline Risk Factor
Randomized Patients

		Treatment Group I			Treatment Group II				
Baseline Risk Factor	Total n	n	Event	Median (95% CI) mo	n	Event	Median (95% CI) mo	Hazard Ratio (95% CI)	Favors TRT Favors Control
All Patients	1188	592	368	11.1 (9.4 - 12.7)	596	334	13.8 (11.7 - 15.9)	0.737 (0.597 - 0.909)	
FISH+/IHC0	124	76	48	7.2 (5.5 - 15.1)	48	30	10.6 (5.4 - 12.6)	0.920 (0.481 - 1.760)	
FISH+/IHC1+	142	66	42	10.2 (6.8 - 13.8)	76	56	8.7 (6.1 - 11.1)	1.246 (0.705 - 2.203)	
FISH+/IHC2+	320	160	106	10.8 (6.8 - 13.0)	160	102	12.3 (9.2 - 15.7)	0.754 (0.511 - 1.111)	
FISH+/IHC3+	524	258	156	12.3 (9.8 - 14.5)	266	122	17.9 (15.7 - 20.7)	0.577 (0.412 - 0.808)	
FISH-/IHC3+	32	12	8	17.7 (1.6 - 20.1)	20	10	17.5 (3.7 - 25.3)	0.826 (0.202 - 3.383)	
FISH no result/IHC3+	32	16	6	19.9 (5.1 - 19.9)	16	6	NE (5.4 - NE)	0.631 (0.125 - 3.181)	
FISH+/IHC no result	14	4	2	7.8 (NE - NE)	10	8	3.8 (1.1 - 12.6)	1.589 (0.159 - 15.930)	

CI = Confidence interval.
Median duration of survival was estimated from Kaplan-Meier curves.

Fig. 1

From the graph above, you can see the last two columns essentially report the same information. However, instead of reading the Odds Ratio and %95 CI from the text column, one can very easily spot the location of the bubble relative to the reference line 1 on the dynamic data visualization which enables you to quickly assess which sub-group of patients showed the benefit of taking treatment.

EXAMPLE 2. CREATE A BOXPLOT IN A SUMMARY REPORT

If, instead of having Odds Ratio as your statistical summary, you have MEAN, STD, Q1 and Q3 or those types of descriptive statistics, a box plot could be more appropriate. Therefore, by changing “seriesplot” to “boxplotparm” in the template (see code below,) you can generate a table with a box plot in it.

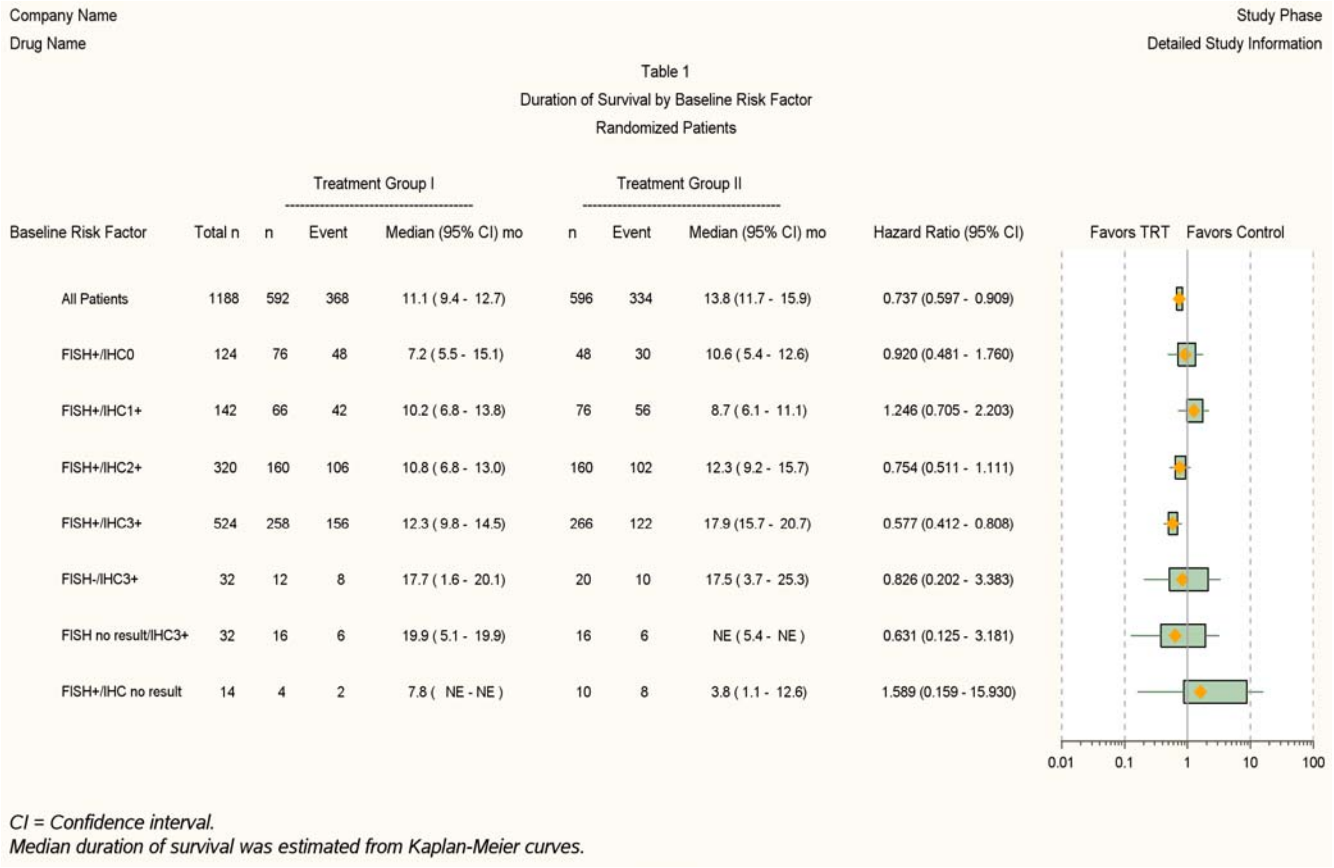
The boxplotparm statement below specifies that the box plot be drawn horizontally and stat=stat indicates which variable in the data set contains the statistics used for the graph.

```
       boxplotparm x=keyvargp y=value stat=stat / orient=horizontal display=(fill notches);
```

The scatter plot here is used for Odds Ratio to improve the readability.

```
       scatterplot x=hazardratio y=her2 / markerattrs=(color=orange symbol=diamondfilled 
size=2pct);
```

Here is the output of example 2 (Fig. 2).



Company Name

Drug Name

Study Phase

Detailed Study Information

Table 1

Duration of Survival by Baseline Risk Factor

Randomized Patients

		Treatment Group I			Treatment Group II				
Baseline Risk Factor	Total n	n	Event	Median (95% CI) mo	n	Event	Median (95% CI) mo	Hazard Ratio (95% CI)	Favors TRT Favors Control
All Patients	1188	592	368	11.1 (9.4 - 12.7)	596	334	13.8 (11.7 - 15.9)	0.737 (0.597 - 0.909)	
FISH+/IHC0	124	76	48	7.2 (5.5 - 15.1)	48	30	10.6 (5.4 - 12.6)	0.920 (0.481 - 1.760)	
FISH+/IHC1+	142	66	42	10.2 (6.8 - 13.8)	76	56	8.7 (6.1 - 11.1)	1.246 (0.705 - 2.203)	
FISH+/IHC2+	320	160	106	10.8 (6.8 - 13.0)	160	102	12.3 (9.2 - 15.7)	0.754 (0.511 - 1.111)	
FISH+/IHC3+	524	258	156	12.3 (9.8 - 14.5)	266	122	17.9 (15.7 - 20.7)	0.577 (0.412 - 0.808)	
FISH-/IHC3+	32	12	8	17.7 (1.6 - 20.1)	20	10	17.5 (3.7 - 25.3)	0.826 (0.202 - 3.383)	
FISH no result/IHC3+	32	16	6	19.9 (5.1 - 19.9)	16	6	NE (5.4 - NE)	0.631 (0.125 - 3.181)	
FISH+/IHC no result	14	4	2	7.8 (NE - NE)	10	8	3.8 (1.1 - 12.6)	1.589 (0.159 - 15.930)	

CI = Confidence interval.

Median duration of survival was estimated from Kaplan-Meier curves.

Fig. 2

The extreme end points of the plot lines are the MIN and MAX while the two ends of the box indicate Q1 and Q3. From the graph above you can quickly visualize how your data is distributed in each individual category rather than interpreting the numbers in the corresponding text column. Note that the width of the boxes can be adjusted to be proportional to any variable you desire.

ADVANTAGES vs. LIMITATIONS

ADVANTAGES

- You don't need to know anything about GTL details to create statistical graphs.
- It is very easy to understand, therefore can be easily modified to reproduce different graphs.
- The quality of graphics is good.
- Save time on creating the same output over the old version of SAS with annotated dataset.
- Platform independent.

LIMITATIONS

- The label is not wrapable. If it is too long for the space allocated, it gets truncated.
- SAS does not accept variable values for the graph attributes yet, so a work around may be needed.
- Not easy to add filler between lines for better readability.
- With size limitation on one page, you may have to break report into pages by filtering data with SGRENDER procedure or in a data step for this type of output.

CONCLUSIONS

With the new features that ODS Graphs and GTL have to offer, a high quality summary/graphics output can be created on the fly. It not only cuts down the program developing time significantly, it helps with the data interpretation efficiency. In the end, we can make our job more efficient by creating fewer outputs with higher quality.

REFERENCES

- Forest Plot from SAS. http://support.sas.com/kb/35/143.html
- SAS documentation v9.2
- GTL (Graphics Template Language) in SAS 9.2. http://www.phuse.eu/download.aspx?type=cms&docID=540
- A Programmer's Introduction to the Graph Template Language. http://www2.sas.com/proceedings/sugi31/262-31.pdf

- Gaining Power from GTL and ODS Style to Control Graphical Output. http://www2.sas.com/proceedings/forum2007/092-2007.pdf
- ODS Statistical Graphics for Clinical Research. http://www2.sas.com/proceedings/sugi31/095-31.pdf

ACKNOWLEDGEMENT

Lelia McConnell and Shannon Moore from SAS support
Vivian Ng: Biostat at Genentech
Graphics club: numbers at Genentech
Mitra Mehrban: SPA Manager

CONTACT INFORMATION

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

Qinghua (Kathy) Chen
Genentech Inc
1 DNA way,
South San Francisco, CA, 94080
Work Phone: 650-225-8427
Email: chen.kathy@gene.com

APPENDIX

```
*************************************************;
*** Name: g_blobbogram.sas                    ***;
*** Purpose: Create blobbogram table with GTL ***;
*** Author: Kathy Chen                        ***;
*** Date: Sept 17 2010                        ***;
*************************************************;
%let dsets=blobbogram;
%init;

title;
options nodate nonumber;

%macro creatmp(ds=,keyvar=);

proc sql;
     select count(distinct(&keyvar)) into :nobs from &ds;
quit;
%put count number: &nobs;

data main;
     set &ds;
     where &keyvar ne ' ';
     constant=1;
run;

*****************************************************;
*** convert data into vertical format for boxplot ***;
*****************************************************;
data temp(keep=&keyvar hazardratio hrlowercl hruppercl ord);
     set &ds;
     ord=_N_;
     where &keyvar ne ' ';
run;

proc transpose data=temp out=temp1 name=stat
     prefix=c;
     by ord &keyvar;
     id ord;
     var hrlowercl hazardratio hruppercl;
run;

%do i=1 %to &nobs;
    data t&i(keep=&keyvar stat c&i c v rename=(&keyvar=keyvargp c&i=value));
         set temp1;
         length c v $2;
         where c&i ne .;
         c='c'||"&i";
         v='v'||"&i";
    run;

    proc datasets;
         append base=box data=t&i;
    run;
%end;

*****************************************************;
*** convert data into horizontal for scatter plot ***;
*****************************************************;
data temp(keep=&keyvar hazardratio ord);
     set &ds;
     ord=_N_;
     where &keyvar ne ' ';
run;

proc transpose data=temp out=temp1(drop=_NAME_ _LABEL_)
```

```
     prefix=c;
     id ord;
     var hazardratio;
run;

proc transpose data=temp out=temp2(drop=_NAME_ _LABEL_)
     prefix=v;
     id ord;
     var &keyvar;
run;

***************************************************;
*** create data for table with blobbogram in it ***;
***************************************************;
data fortable;
     merge box main temp1 temp2;
run;

proc print;
run;

*********************************************;
*** create macro variables for blobbogram ***;
*********************************************;
proc sql;
     select sizen*4 into :s1 - :s%left(&nobs) from fortable;
quit;
run;

****************************************;
*** create the table layout template ***;
****************************************;
proc template;
  define statgraph ForestPlot ;
  nmvar %do i=1 %to &nobs;
            s&i
        %end;;
  begingraph / designwidth=1000px designheight=600px;
  entrytitle halign=left "Company Name" halign=right "Study Phase" /  pad=(bottom=5px);
  entrytitle halign=left "Drug Name" halign=right "Detailed Study information" /
pad=(bottom=5px);
  entrytitle "Table 1" /  pad=(bottom=5px);
  entrytitle "Duration of Survival by HER2 Status" /  pad=(bottom=5px);
  entrytitle "Randomized Patients" /  pad=(bottom=5px);
  entrytitle "                   " /  pad=(bottom=5px);
  entrytitle halign=left "
Treatment Group I                                               Treatment Group II"
/pad=(bottom=0px);

  entrytitle halign=left "
--------------------------------------                           ----------------------------
-----------" /pad=(bottom=0px);
  layout lattice / columns=10   columngutter=0
                   columnweights=(.14 .04 .04 .05 .14  .04 .05 .14 .15 .21);

  layout overlay / walldisplay=none border=false
                   y2axisopts=(reverse=true type=discrete display=(tickvalues) offsetmin=0.09
offsetmax=0.09)
                   xaxisopts=(display=none offsetmin=0 offsetmax=0);
    entry halign=left "Baseline Risk Factor"  / textattrs=GraphLabelText location=outside
valign=top;
    scatterplot y=her2 x=constant  / yaxis=y2 markerattrs=(size=0);
  endlayout;
  layout overlay  / walldisplay=none border=false
                    yaxisopts=(reverse=true type=discrete display=none)
                    xaxisopts=(display=none offsetmin=0 offsetmax=0);
    entry "Total n" / location=outside valign=top textattrs=GraphLabelText;
    scatterplot y=her2 x=constant  /   markercharacter=total
                                       markercharacterattrs=GraphDataText;
  endlayout;
  %do i=0 %to 1;
      layout overlay  / walldisplay=none border=false
                    yaxisopts=(reverse=true type=discrete display=none)
                    xaxisopts=(display=none offsetmin=0 offsetmax=0);
       entry "n" / location=outside valign=top textattrs=GraphLabelText;
       scatterplot y=her2 x=constant  /  markercharacter=n&i
                                      markercharacterattrs=GraphDataText;
      endlayout;
      layout overlay  / walldisplay=none border=false
                    yaxisopts=(reverse=true type=discrete display=none)
                    xaxisopts=(display=none offsetmin=0 offsetmax=0);
      entry "Event" / location=outside valign=top textattrs=GraphLabelText;
      scatterplot y=her2 x=constant  /  markercharacter=event&i
                                      markercharacterattrs=GraphDataText;
      endlayout;
      layout overlay  / walldisplay=none border=false
                    yaxisopts=(reverse=true type=discrete display=none)
                    xaxisopts=(display=none offsetmin=0 offsetmax=0);
      entry "Median (95% CI) mo" / location=outside valign=top textattrs=GraphLabelText;
      scatterplot y=her2 x=constant  /  markercharacter=med_ci&i
                                      markercharacterattrs=GraphDataText;
      endlayout;
  %end;
  layout overlay  / walldisplay=none border=false
                    yaxisopts=(reverse=true type=discrete display=none)
                    xaxisopts=(display=none offsetmin=0 offsetmax=0);
    entry "Hazard Ratio (95% CI)" / location=outside valign=top textattrs=GraphLabelText;
    scatterplot y=her2 x=constant  /  markercharacter=hr_ci
                                      markercharacterattrs=GraphDataText;
  endlayout;
  layout overlay / walldisplay=(fill)
                   yaxisopts=( display=none reverse=true  offsetmin=0.1 offsetmax=0.1
                   linearopts=(integer=true))
                   /*xaxisopts=(offsetmin=0 offsetmax=0);*/
                   xaxisopts=(type=log offsetmin=0 offsetmax=0
                label="       "
                   logopts=(base=10 tickintervalstyle=logexpand
                   minorticks=true viewmin=0.01 viewmax=100 ));
    entry "Favors TRT    Favors Control" / location=outside valign=top
textattrs=GraphLabelText;
    *boxplotparm x=keyvargp y=value stat=stat / orient=horizontal display=(fill notches);
    *scatterplot x=hazardratio y=her2 / markerattrs=(color=orange symbol=diamondfilled
size=2pct);
    seriesplot y=keyvargp x=value / group=keyvargp lineattrs=(pattern=solid);
    %do i=1 %to &nobs;
        scatterplot x=c&i y=v&i / markerattrs=(color=black symbol=circle size=s&i);
    %end;;

    referenceline x=0.1 / lineattrs=(pattern=shortdash);
    referenceline x=1  / lineattrs=(pattern=solid);
    referenceline x=10 / lineattrs=(pattern=shortdash);
    referenceline x=.01 / lineattrs=(pattern=shortdash);
    referenceline x=100 / lineattrs=(pattern=shortdash);
  endlayout;
  entryfootnote halign=left "CI = Confidence interval." /  pad=(bottom=0px);
  entryfootnote halign=left "Median duration of survival was estimated from Kaplan-Meier
curves." /  pad=(bottom=0px);
  entryfootnote halign=left "&FN_9L" halign=center "&FN_9C" /  pad=(bottom=5px);
  endlayout;
  endgraph;
  end;
run;
```

```
proc template;
  define Style foreststyle;
    parent = styles.analysis;
    style GraphFonts from GraphFonts
      "Fonts used in graph styles" /
      'GraphTitleFont' = (", ",8pt)
      'GraphLabelFont' = (", ",8pt)
      'GraphValueFont' = (", ",7pt)
      'GraphDataFont' = (", ",7pt);
  end;
run;

***********************************************;
*** run SGRENDER to generate the pdf output ***;
***********************************************;

ods listing image_dpi=300 gpath='../pdf' style=foreststyle;
ods graphics / imagename="&jobname" imagefmt=pdf;
proc sgrender data=fortable template=ForestPlot;
run;
ods listing close;
ods graphics off;

%mend creatmp;

%creatmp(ds=outdata.blobbogram,keyvar=her2);
```