

Customizing Survival Plot Using ODS Graphics Template Language

Fang Dong, Aastrom Biosciences, Ann Arbor, Michigan

ABSTRACT

SAS has made enormous strides in advancing its statistical graphics capabilities over the last several releases of the software. Specifically, starting from SAS[®] 9.1 when the ODS Graphics were experimental, to SAS 9.2 when ODS Graphics became production, and to SAS 9.3 where over 80 procedures now utilize the ODS Graphics capability. It is imperative for the programmer, including the veteran, to keep up with the advancement in order to produce highly customized quality graphs. This paper describes several key language components necessary to produce a highly customized graph. While the example is specific to a survival plot, the concepts should apply to customizing any other graphs.

INTRODUCTION

Prior to ODS Graphics, SAS was not known as the software that produced high quality graphics, let alone producing them easily. As SAS acknowledged, it often took several steps and hundreds of lines of code to produce a satisfactory statistics graph¹. SAS 9.1 implemented an experimental version of the ODS Statistical Graphics technology which was used in two dozen statistical procedures to produce high quality graphs. The ODS Graphics Template Language also enables the users to create highly customized high-quality graphs more easily than older SAS/Graph technology. In SAS 9.2, the ODS Graphics became a production version, meaning the syntax became more stable, and even more SAS products and procedures utilized the technology to produce high quality default graphs. In SAS 9.3, more than 80 statistical procedures produce default graphs with the ODS Graphics functionality. Additionally, there are many enhancements in the Graphics Template Language that make creating highly customized graphs easier than ever. Still, the techniques to create or modify a graph are not intuitive to even a veteran programmer. We will describe a few key components that are necessary in order to create a highly customized graph in SAS 9.3. We will use a survival plot as an example, and the concepts should be applicable to other type of graphs.

BACKGROUND

Kaplan-Meier curves or survival plots are often used to display treatment effects of new medical drugs or therapies. Aastrom has reported that its experimental expanded autologous cell product, ixmyelocel-T, showed promise in treating patients with Critical Limb Ischemia in a phase 2 study⁷. The evidence for the therapy's potential efficacy was shown in a KM plot initially produced from the LIFETEST procedure with ODS Graphics in SAS 9.1.3, see Figure 1.

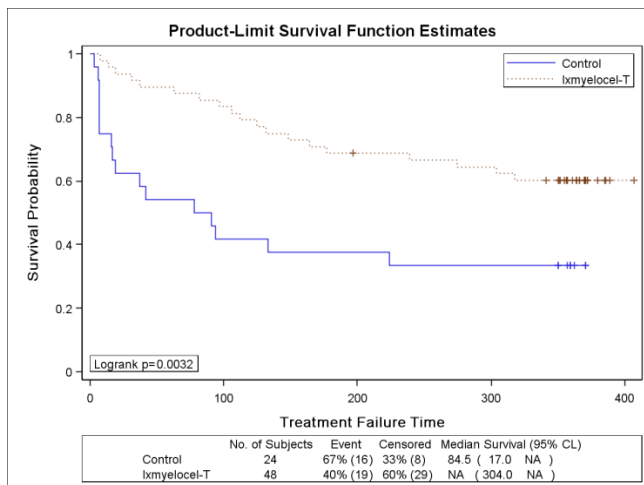


Figure 1 SAS 9.1.3 Default Output with Survival Summary Statistics

There were many requests to modify the plot to meet various publishing guidelines of journals. Typical requests include changing the color, size, resolution, and the addition of an 'at risk' table. The ways of achieving these modifications have evolved from SAS 9.1.3 to 9.3. For example, in SAS 9.1.3, one would need to use the annotation data set and techniques to add the 'at risk' table. In SAS 9.3, one can add an option in PROC LIFETEST to produce

the 'at risk' table automatically, see Figure 2, where we can also see a remarkable visual improvement of the output using SAS 9.3. However, SAS 9.3 no longer produces the survival summary statistics table automatically that was available in SAS 9.1.3 (bottom of Figure 1). In the next sections, we will describe the key components one needs to understand in order to make the modifications, including adding the survival summary statistics table back to the plot using SAS 9.3.

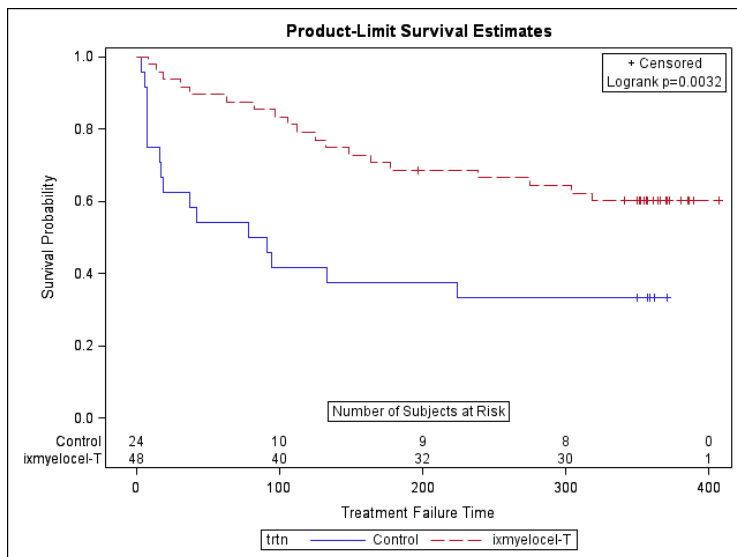


Figure 2 SAS 9.3 Default Output with At Risk Numbers

WAYS TO CUSTOMIZE THE SURVIVAL PLOT IN SAS 9.3

MODIFYING THE SIZE AND RESOLUTION

We will start from the easiest modification. Modifying the size of a graph is achieved in the overall *ODS Graphics On* statement, while modifying the resolution is done through the ODS destination statement. What is amazing is that they are not controlled in the same statement, leaving one with an impression of 'fragmentation'. This is perhaps due to the fact that SAS uses two types of templates, the ODS style templates and the ODS graphics templates, to regulate the appearance of a graph, which we will discuss further in the next section.

```
ods graphics on / width = 3 in;

ods rtf file = "mysurvivalplot.rtf" image_dpi = 300 style=mystyles;
```

MODIFYING THE LINE COLOR AND PATTERN

Here we need to understand a few more points. First, SAS 9.3 uses both ODS style templates and ODS Graphics templates to control the appearance of a graph. The ODS style templates are relevant to the particular ODS destination chosen, such as pdf, rtf, HTML, etc. The ODS Graphics templates are specific to each graph within individual statistical procedures, such as the survival plot within the PROC LIFETEST. Second, SAS 9.3 supplies more ODS styles that can meet many line color and pattern preferences. However, if one wants to customize beyond those templates that are supplied, one will need to modify either the ODS style template, the ODS Graphics template, or both that are used to produce the graph. Third, while one can modify the line colors and patterns for ungrouped data within the graphics template, one cannot yet modify the line colors and patterns for grouped data within the graphics template alone in SAS 9.3^{3,6}.

Since our data has two groups, the control group and the ixmyelocel-T group, we need to modify the style template that is used for the ODS destination. Below is an example to change the line colors to black and also change the line styles, where ABI_Landscape_10pt is a customized style template that is used in Aastrom as a corporate standard. The ABI_Landscape_10pt style template is adapted from *styles.rtf* supplied by SAS.

```
proc template;
  define style Styles.mystyles;
    parent = styles.ABI_Landscape_10pt;
    style GraphData1 /
      contrastcolor = black
      linestyle = 1;          /**** Solid Line ****/
    style GraphData2 /
      contrastcolor = black
      linestyle = 3;          /**** Short Dash Line ****/
  end;
run;
```

One invokes this newly customized style in the ODS destination statement, as in the example below.

```
ods rtf file = "mysurvivalplot.rtf" image_dpi = 300 style=mystyles;
```

ADDING THE SURVIVAL SUMMARY STATISTICS TABLE

To do this, we need to learn some of the basics of the Graphics Template Language. Fortunately, there are several good references to get one started. Specifically, reference no.3 provides a good overview on how to customize various aspects of the survival plot while reference no.4 provides a detailed example on how to display the survival summary statistics table.

Here we will summarize the approaches that were used to create the customized output in Figure 3 with SAS 9.3, based on the learning from the references. Reference no. 5 showed how to create a template that is easier to modify than the default survival plot template installed in the PROC LIFETEST procedure. The modified template as shown in Display 1 organizes the code into distinct modules and makes it easier to locate and change the code necessary to effect a change in the output. However, we need not re-create such a modified template, since SAS has readily supplied it in the Samples. Instead, we need to understand the approach generally as outlined in reference no.5 and retrieve the template from the SAS Sample inventory.

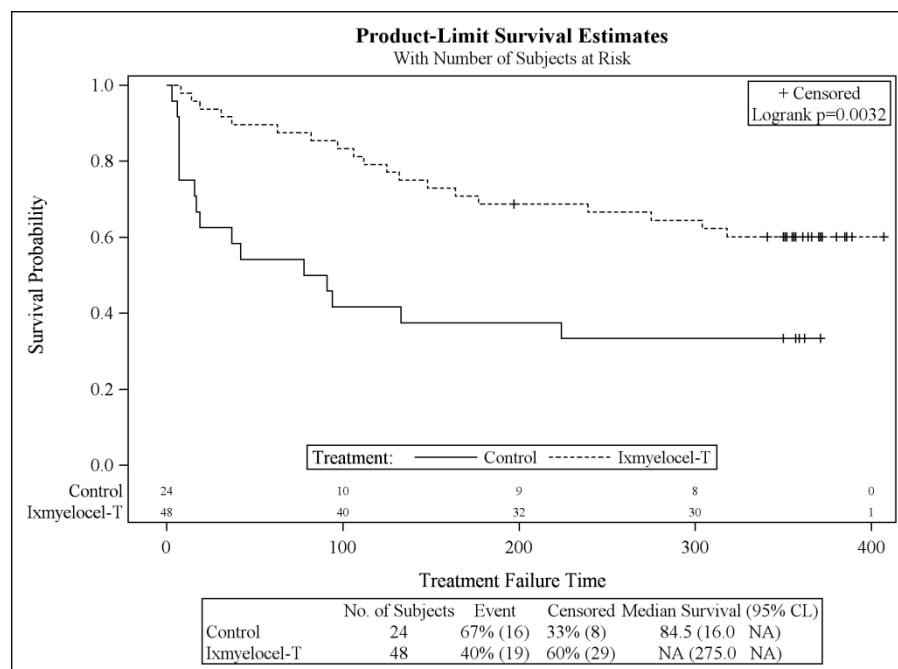


Figure 3 SAS 9.3 Customized Output with At Risk Numbers and Survival Summary Statistics

```

%macro SurvivalTemplateRestore;
  /**Declare and define global macro variables***/
1 %global TitleText0 TitleText1 TitleText2 yOptions xOptions tips
   groups bandopts gridopts blockopts censored censorstr;
   ...../**Skipped Code***/
2 %macro SurvivalTemplate;
   proc template;
     define statgraph Stat.Lifetest.Graphics.ProductLimitSurvival;
     ...../**Skipped Code that declare dynamic variables***/
     BeginGraph;
       if (NSTRATA=1)
         ..... /**Skipped Code***/
         layout overlay / xaxisopts=(&xoptions) yaxisopts=(&yoptions);
         /**Calling the Macro to plot single stratum ***/
         %singlestratum
         endlayout;
       else
         ..... /**Skipped Code***/
         layout overlay / xaxisopts=(&xoptions) yaxisopts=(&yoptions);
         /**Calling the Macro to plot multiple strata ***/
         %multiplestrata
         endlayout;
       endif;
     EndGraph;
   end;
   run;
%mend;
3 %macro entry_p;
  /**Skipped Code that defines how to display the p-value***/
%mend;
4 %macro SingleStratum;
  /**Skipped Code that defines the macro to plot single stratum***/
  .....
%mend;
5 %macro MultipleStrata;
  /**Skipped Code that defines the macro to plot multiple strata***/
  .....
  DiscreteLegend "Survival" / title=GROUPNAME location=outside;
  /**More Skipped Code that defines the macro to plot multiple strata***/
  .....
%mend;
6 %SurvivalTemplate
%mend;

```

Display 1 SAS Supplied and Modified PROC LIFETEST Survival Plot Template

Before we go further, let us describe the key components of the SAS-supplied modified survival plot template. Display 1 shows a skeleton of this template encompassed within the macro `%SurvivalTemplateRestore`. This macro can be roughly divided into six parts.

Part 1 declares global macro variables and defines their values.

Part 2 defines the modified survival plot template in the macro `%SurvivalTemplate` where it calls two other macros `%SingleStratum` and `%MultipleStrata`.

Part 3 defines how to display the p values in the macro `%entry_p` that is called by the `%SingleStratum` and `%MultipleStrata` macros.

Parts 4 and 5 define how the single stratum and multiple strata survival plots are displayed respectively. Details of the code are not included in the display so that only the major functions of the template are highlighted. Please refer to reference no.5 for the complete code.

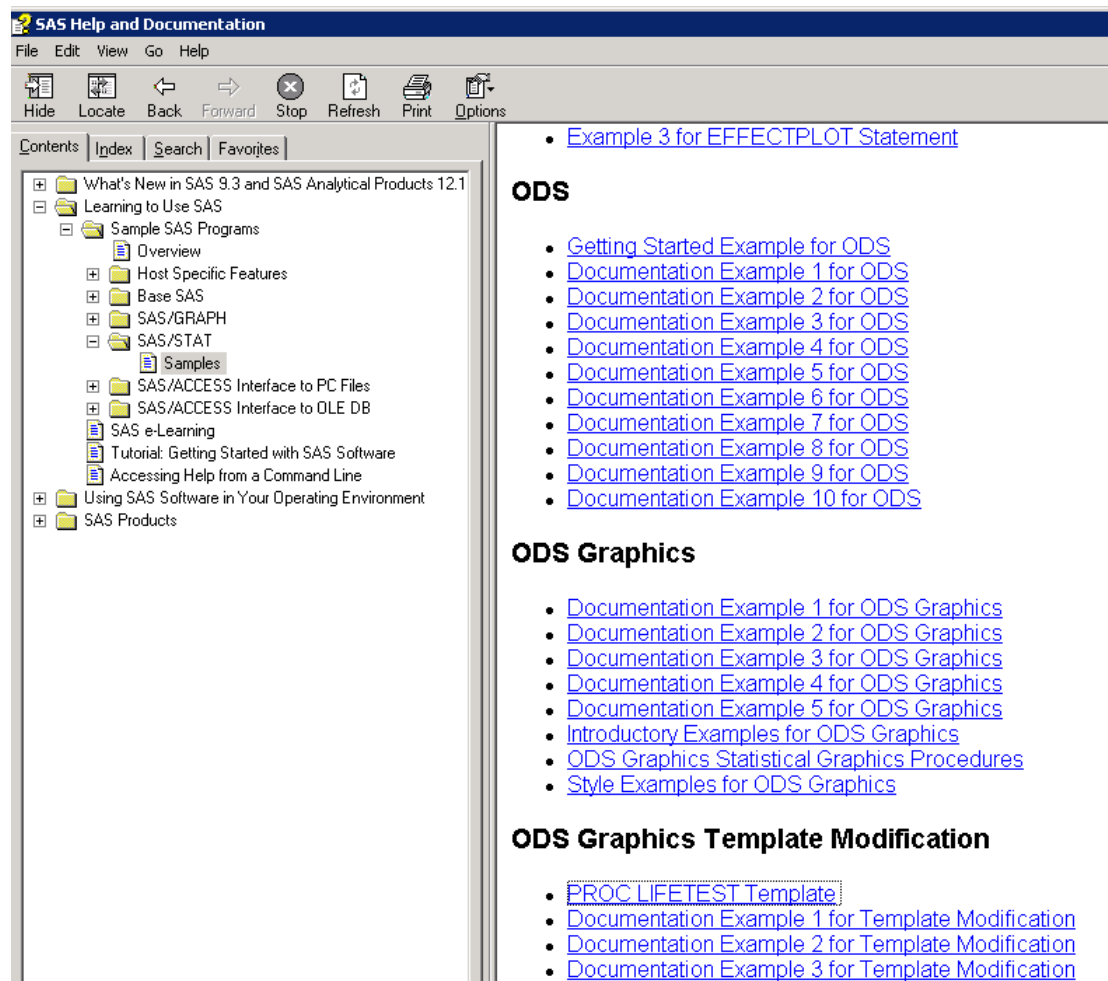
Part 6 simply calls the modified survival plot template macro defined in part 2 and unless otherwise specified, the template will be stored in the SAS *User* area. Later we will discuss how to make SAS use this customized survival plot template in the SAS *User* area instead of the SAS installed default survival plot template.

Please also note the highlighted code in part 5 where the legend location is displayed outside the plot by default as shown in Figure 2. We will modify it to have the legend displayed inside the plot (Figure 3) in step 5 below.

Now that we have gained a general understanding of this template that is indeed easier to modify, we can use the following steps to add the *Event Summary Statistics* table to the survival plot.

Step 1: Copy the SAS supplied sample code of the modified PROC LIFETEST Survival Plot template to your local directory

Reference no.5 provides details on how to accomplish this step. Specifically, in an interactive SAS session, click on the *Help* icon, that will open the SAS Help and Documentation window with two panels. On the left panel, click on 'Learning to Use SAS', then 'Sample SAS Programs', then 'SAS/STAT', then 'Samples'. At this point, click on 'PROC LIFETEST TEMPLATE' on the right panel under the 'ODS Graphics Template Modification' category, as show in Display 2. You will find the modified and modulated 'ProductLimitSurvival' template encompassed inside the *%SurvivalTemplateRestore* macro. The skeleton of this macro is shown in Display 1. Save this macro in your user defined macro library location so you can use it later in your programs.



Display 2 Retrieve the Modified PROC LIFETEST Survival Plot Template from SAS Samples

Step 2: Define the template search path so the user modified template will be used by PROC LIFETEST instead of the default template supplied by SAS

```
Ods path SASUSER.TEMPLAT(write) SASHELP.TMPLMST(read);
```

When there are multiple templates with the same name in different template stores, SAS will use the template in the leftmost template store as defined in the above statement. Here the SAS supplied default survival template 'ProductLimitSurvival' is in the *SASHELP.TMPLMST* store while the user modified survival template with the same name will be in the *SASUSER.TEMPLAT* store when we have finished all the steps below and will be the one that is used to display the survival plot as shown in Figure 3.

Step 3: Call the macro obtained from step 1

```
%SurvivalTemplateRestore
```

This will make the SAS supplied modified survival template macros and macro variables as shown in Display 1 available to use in your SAS program.

Step 4: Create the Event Summary Statistics table with Graphics Template Language

Here we expanded the example from reference no.4. First, make the table header with the %header macro which will then be called in the %table2 macro. Next make the table with the macro %table2. Display 3 shows the respective macros with explanations in the comments.

Keeping in mind that the table of the Survival Summary Statistics is what we want to add, it has five columns as shown in Figure 4. Particularly, column 1 is the treatment group or the strata and has no heading. Column 2 is the number of subjects in each group and column 3 is the number of subjects who experienced an event in each group. Column 4 is the number of subjects who did not experience any event and were censored at the end of the study. Column 5 is the median survival time with the 95% confidence interval in the brackets. NA means not available.

	Column 2	Column 3	Column 4	Column 5
Column 1	No. of Subjects	Event	Censored	Median Survival (95% CL)
Control	24	67% (16)	33% (8)	84.5 (16.0 NA)
Ixmyelocel-T	48	40% (19)	60% (29)	NA (275.0 NA)

Figure 4 Survival Summary Statistics Table for the KM Plot

```

%let fmt = bestd6.;
%macro header; /** To make column headers ***/
  entry halign=center "No. of Subjects"; /*Column 2 header*/
  entry halign=center "Event"; /*Column 3 header*/
  entry halign=center " "; /**Dummy column to add space between column 3 and 4 to improve
  the appearance***/
  entry halign=center "Censored"; /*Column 4 header*/
  entry halign=center "Median Survival ( " PctMedianConfid " CL)"; /*Column 5 header*/
%mend;
%macro table2;
  columnheaders;
  layout overlay / pad=(top=5);
  layout gridded / columns=6 border=TRUE;
  dynamic PctMedianConfid;
  entry " "; /**Column 1 has no header***/
  %header
  %do i = 1 %to 6;
    dynamic StrVal&i NObs&i NEvent&i Median&i
      LowerMedian&i UpperMedian&i; /**Summary Statistics***/
    if (&i <= nstrata)
      entry halign=left StrVal&i; /**Column 1***/
      entry halign=center NObs&i; /**Column 2***/
      entry halign=center eval(strip(put(100*(NEvent&i/NObs&i),6.0))||'%'
('||strip(put(NEvent&i,best.)||')'); /**Column 3***/
      entry " "; /**Dummy column to add space between column 3 and 4 to improve
      the appearance***/
      entry halign=left eval(strip(put(100*((NObs&i-NEvent&i)/NObs&i),6.0))||'%'
('||strip(put((NObs&i-NEvent&i),best.)||')'); /**Column 4***/
      if (Median&i ne . and UpperMedian&i ne . and LowerMedian&i ne .)
        entry halign=center eval(strip(put(Median&i,&fmt))||'
('||strip(put(LowerMedian&i,&fmt))||' '||strip(put(UpperMedian&i,&fmt))||')');
      endif;
      if (Median&i ne . and LowerMedian&i ne . and UpperMedian&i eq .)
        entry halign=center eval(strip(put(Median&i,6.1))||'
('||strip(put(LowerMedian&i,6.1))||' NA)');
      endif;
      if (Median&i eq . and LowerMedian&i ne . and UpperMedian&i ne .)
        entry halign=center 'NA (' eval(strip(put(LowerMedian&i,6.1))||'
'|strip(put(UpperMedian&i,6.1))||')');
      endif;
      if (Median&i eq . and LowerMedian&i ne . and UpperMedian&i eq .)
        entry halign=center 'NA (' eval(strip(put(LowerMedian&i,6.1))||' NA)');
      endif;
    endif;
  %end;
  %end;
  endlayout;
  endlayout;
  endcolumnheaders;
%mend;

```

All highlighted code is to create column 5 in Figure 4.

Display 3 Customized Macros to Create the Survival Summary Statistics Table

Step 5: Modify the %MultipleStrata macro to display the treatment group legend inside the plot

```

%macro MultipleStrata;
  stepplot y=SURVIVAL x=TIME / &groups name="Survival" &tips;

  if (PLOTCEASURED)
    scatterplot y=CENSORED x=TIME / &groups &censored &tipl;
  endif;

  if (PLOTATRISK=1)
    innermargin / align=bottom;
    blockplot x=TATRISK block=ATRISK / class=CLASSATRISK
      display=(label values) &blockopts;
    endinnermargin;
  endif;

  DiscreteLegend "Survival" / title='Treatment:' location=inside autoalign=(Bottom);

  if (PLOTCEASURED)
    if (PLOTTEST)
      layout gridded / rows=2 &gridopts;
      entry &cursorstr;
      %entry_p
    endlayout;
    else
      layout gridded / rows=1 &gridopts;
      entry &cursorstr;
    endlayout;
  endif;
  else
    if (PLOTTEST)
      layout gridded / rows=1 &gridopts;
      %entry_p
    endlayout;
  endif;
  endif;
%mend;

```

Display 4 Customized Macro to Display the Legend Inside the Plot

In Figure 2, we can see that the legend by default is outside the plot and at the bottom of the graph. We need the space to put the summary statistics table there, hence we need to move the legend inside the plot.

Please note that only the statements needed for our customized plot are included in the abbreviated version of the %MultipleStrata shown in Display 4.

Step 6: Modify the %SurvivalTemplate macro to call the customized %MultipleStrata macro created in the step 5 above and add the Survival Summary Statistics table created in step 4

In the underlined text inside Display 5, note that we are now using the *lattice layout*, instead of the *overlay layout* as used in part 2 of Display 1 (see the underlined text). This is because we want to add the table below the survival plot. The lattice layout adds additional space to displace the summary statistics table.


```

%macro SurvivalTemplate;
  proc template;
    define statgraph Stat.Lifetest.Graphics.ProductLimitSurvival;
      dynamic NStrata xName plotAtRisk plotCL plotHW plotEP labelCL
        %if %nrbquote(&censored) ne %then plotCensored;
      labelHW labelEP maxTime xtickVals xtickValFitPol method StratumID
      classAtRisk plotBand plotTest GroupName yMin Transparency SecondTitle
      TestName pValue;
      BeginGraph / designheight=defaultdesignwidth;

        entrytitle &titletext2;
        if (EXISTS(SECONDTITLE))
          entrytitle SECONDTITLE / textattrs=GRAPHVALUETEXT;
        endif;

        layout lattice / rows=1 columns=1;
        layout overlay / xaxisopts=(&xoptions) yaxisopts=(&yoptions);
        %multiplestrata
        endlayout;
        %table2
        endlayout;

      EndGraph;
    end;
  run;
%mend;

```

Display 5 Customized Macro that Calls the Customized %MultipleStrata and %table2 Macros

Step 7: Call %SurvivalTemplate created in step 6

```
%SurvivalTemplate;
```

This will create and store the final *ProductLimitSurvival* template in the *SASUSER.TEMPLAT* store as defined in step 2 above.

Step 8: Put them all together

PROC LIFETEST in Display 6 will use the new *ProductLimitSurvival* template created from the previous steps to produce the survival plot. Note that we used the *test* option to output the p-value and the *atrisk* option to display the at risk table.

```

ods rtf file = "mysurvivalplot.rtf" image_dpi = 300 style=mystyles;
ods graphics on / width = 6.75 in height = 5 in;
ods noptitle;
proc lifetest data=adef plots=survival(test atrisk=0 to 400 by 100);
  time trtft*trtf(1);
  strata trtn;
  format trtn trt.;
run;

```

Display 6 PROC LIFETEST Code to Produce the Plot as Shown in Figure 3

CONCLUSION

With the advent of ODS Graphics and ODS Graphics Template Language, creating highly sophisticated statistical graphs in SAS 9.3 is more attainable than ever before. However, there are a few pieces of the puzzle that one needs to know in order to assemble a beautiful graph. The key pieces are the *ODS Graphics On* statement, the ODS destination statement, the ODS style template for the chosen destination, and the ODS graphics template used in the particular statistical procedure. Once we are aware of the pieces, we can use the ample references to guide us to create a customized graph to suit our fancy!

REFERENCES

1. Robert N. Rodriguez and Tonya E. Balan (2006), 'Creating Statistical Graphics in SAS 9.2: What Every Statistical User Should Know', Proceedings of the Thirty-first Annual SAS Users Group International Conference. Cary, NC: SAS Institute Inc.
2. Robert N. Rodriguez (2011), 'An Overview of ODS Statistical Graphics in SAS 9.3', available at http://support.sas.com/resources/papers/76822_ODSGraph2011.pdf
3. ODS Graphics Template Modification: Customizing Survival Plots - 9.3
http://support.sas.com/documentation/cdl/en/statug/63962/HTML/default/viewer.htm#statug_templt_a000000005_9.htm
4. ODS Graphics Template Modification: Displaying Survival Summary Statistics - 9.3
http://support.sas.com/documentation/cdl/en/statug/63962/HTML/default/viewer.htm#statug_templt_a000000006_9.htm
5. ODS Graphics Template Modification: Creating a Template That is Easy to Modify - 9.3
http://support.sas.com/documentation/cdl/en/statug/63962/HTML/default/viewer.htm#statug_templt_a000000006_2.htm
6. SAS(R) 9.3 Graph Template Language: Reference, Third Edition
<http://support.sas.com/documentation/cdl/en/grstatgraph/65377/HTML/default/viewer.htm#p0icsws0thpagn1t12m27fyqjic.htm>
7. Powell RJ, Marston W, Berceli S, Guzman R, Henry TD, Longcore AT, et al. Cellular therapy with ixmyelocel-T to treat critical limb ischemia: the randomized, double-blind, placebo-controlled RESTORE-CLI trial. Mol Ther 2012 Jun;20(6):1280-6

ACKNOWLEDGMENTS

I wish to thank Aastrom Biosciences for providing me the opportunity to work on this paper. Mr. Rich Synowiec, Mr. Don Sizemore and Dr. Caryn Cramer all kindly read the draft paper and provided valuable comments. All remaining errors are solely my responsibility.

I also wish to thank the staff from SAS Institute Inc. for answering my numerous questions from SAS 9.1.3 to SAS 9.3. They are the best in customer support!

CONTACT INFORMATION

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

Fang Dong
Aastrom Biosciences
Lobby K, 24 Frank Lloyd Wright Dr.
Ann Arbor, MI 48108
Work Phone: 734-418-4426
E-mail: fdong@aastrom.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

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