PharmaSUG 2015 - Paper DV10

# Graphical Presentation of Clinical Data in Oncology Trials

Murali Kanakenahalli, Seattle Genetics, Inc., Bothell, WA
Avani Kaja, Seattle Genetics, Inc., Bothell, WA

## ABSTRACT

Graphs are gaining ground in usage for presenting data in oncology clinical trials. As the options in ODS graphics and Graph Template Language (GTL) have grown over the last few years, so has the demand for details in graphs. Improvements in ODS and GTL have also provided an opportunity to create various types of graphs with ease. This paper discusses different kinds of graphs like KM plots, waterfall plots, swimmer plots, etc. which are used to present data on survival, adverse events, and response. The paper also showcases our attempts to use the standardized ODS template for all the plots along with Graph Template Language templates.

## INTRODUCTION

Clinical trials generate a large and diverse amount of data. To analyze these complex data better a concise visual presentation is very much essential. High quality graphs help us analyze and understand the data from different clinical trial domains. The graphs have to be well designed and carefully created to represent analysis data. Statistical Graphics (SG) and Graph Template language (GTL) provide many possible combinations of statements to create graphs. GTL has also made the task of including derived statistics values in the graph easier.

The goal of this paper is to discuss visual presentation of data in oncology clinical trials. We will show some sample efficacy data graphs, safety data graphs, and suggest some useful options to help customize the graphs.

Commonly generated graphs for the clinical trials include Kaplan Meier plots, scatter plot, bar chart, and line plots. Examples of these are shown below.

### Kaplan Meier Graph

Kaplan-Meier estimate is one of the best options to be used to measure the fraction of subjects living for a certain amount of time after treatment[4]. In oncology clinical trials the effect of an intervention is often assessed by measuring the number of subjects survived or free of progression after that intervention over a period of time[4]. The Kaplan Meier Graph below was generated using ODS template with GTL graph template. This plot shows the survival rate by time in weeks for two treatment groups. The summary statistics can be displayed within the graph.

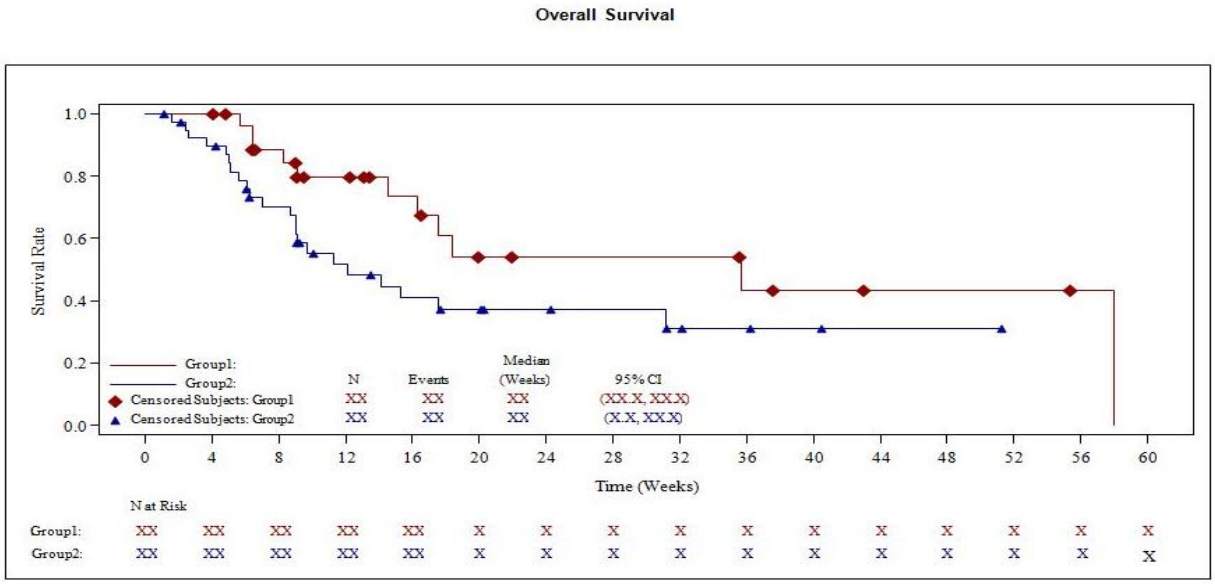



Figure 1. Kaplan Meier Graph with two treatment groups.

ODS Template for the Figure 1 is given below:

```
proc template;
define style sty_km;
parent=rtftnr10;
style graphfonts from graphfonts /
      'GraphLabelFont'=('times new roman',10pt)
      'GraphTitleFont'=('Arial',11pt)
      'GraphFootnoteFont'=('times new roman',10pt);
/* defining the linestyle and marker symbol options */
style graphdata1 from graphdata1/contrastcolor=darkred
                                 linestyle=1;
style graphdata2 from graphdata1/contrastcolor=darkblue
                                 linestyle=1 ;
style graphdata3 from graphdata1 /markersymbol='diamondfilled'
                                 contrastcolor=darkred;
style graphdata4 from graphdata1 /markersymbol='trianglefilled'
                                  contrastcolor=darkblue;
end;
run;
```

We have tried to breakdown the code explaining every option used in the code.

- GraphLabelFont – Axis labels can be modified using this option.
- GraphTitleFont – This is used for titles within the graph.
- GraphFootnoteFont – This is used for footnotes within the graph.
- Graphdata1 - Graphdata8 can be used with different marker symbols, line styles, and line color.

  Since there are 2 treatment groups in Figure 1, we are using 4 separate graphdata statements to define the linestyles and marker symbol separately. This synchronizes with the plot statements used in the GTL graph template. The reason to use linestyle before marker symbols is because graph template has step plot as the first plot statement and scatter plot as the second plot statement.

GTL graph template for the Figure1 is given below:

```
proc template;
define statgraph km_ar;
mvar  c1r0 c2r0 c3r0 c4r0 c5r0
       c1r01 c2r01 c3r01 c4r01 c5r01
       c1r1 c1r2 c2r1 c2r2 c3r1 c3r2 c4r1 c4r2 c5r1 c5r2 test;
begingraph / designwidth=9in designheight=4.5in ;
entrytitle 'Sample Graph';
entryfootnote 'Sample footnote';
layout lattice / rowweights=(0.85 .05 .1);

/* Defining the y-axis, x-axis and plot statement*/

layout overlay / yaxisopts=(label='Survival Rate' linearopts=(viewmin=0 viewmax=1))
                 cycleattrs=true
                 xaxisopts=(label="Time (Weeks)"  offsetmin=0.01 offsetmax=0.01
                 linearopts=(viewmin=0 tickvaluelist=(&xvals.)));
stepplot x=aval y=survival /  group=&strat name='km'lineattrs=(thickness=2pt) ;
scatterplot x=eval(choosen(_censor_+1,.,aval)) y=survival/
             group=trt2pn name='cen' legendlabel="Censored Subjects";
/*Defining the Legends*/
discretelegend 'km'  / location=inside halign=left pad=(right=190 bottom=30
                       valign=bottom across=1 border=false valueattrs=(size=8pt);
discretelegend 'cen' / location=inside halign=left valign=bottom
                       order=columnmajor border=false valueattrs=(size=8pt);

/*Positioning the dynamic variables within the graph*/

layout gridded /valign=bottom halign=left border=false opaque=false columns=5
                order=rowmajor ;
entry halign=center c1r0 / pad=(left=80) textattrs=(size=8pt color=black);
entry halign=center c2r0 / pad=(left=80) textattrs=(size=8pt color=black);
```

```
entry halign=center c3r0 / pad=(left=30) textattrs=(size=8pt color=black) ;
entry halign=center c4r0 / pad=(left=30) textattrs=(size=8pt color=black) ;
entry halign=center c5r0 / pad=(left=30) textattrs=(size=8pt color=black) ;
entry halign=center c1r01 / pad=(left=80) textattrs=(size=8pt color=black) ;
entry halign=center c2r01 / pad=(left=80) textattrs=(size=8pt color=black) ;
entry halign=center c3r01 / pad=(left=30) textattrs=(size=8pt color=black) ;
entry halign=center c4r01 / pad=(left=30) textattrs=(size=8pt color=black) ;
entry halign=center c5r01 / pad=(left=30) textattrs=(size=8pt color=black) ;
entry halign=center c1r1 / pad=(left=80) textattrs=(size=8pt color=darkred);
entry halign=center c2r1 / pad=(left=80) textattrs=(size=8pt color=darkred);
entry halign=center c3r1 / pad=(left=30) textattrs=(size=8pt color=darkred);
entry halign=center c4r1 / pad=(left=30) textattrs=(size=8pt color=darkred);
entry halign=center c5r1 / pad=(left=30) textattrs=(size=8pt color=darkred);
entry halign=center c1r2 / pad=(left=80) textattrs=(size=8pt color=darkblue);
entry halign=center c2r2 / pad=(left=80) textattrs=(size=8pt color=darkblue);
entry halign=center c3r2 / pad=(left=30) textattrs=(size=8pt color=darkblue);
entry halign=center c4r2 / pad=(left=30) textattrs=(size=8pt color=darkblue);
entry halign=center c5r2 / pad=(left=30) textattrs=(size=8pt color=darkblue);
endlayout;
endlayout;

/*Positioning the 'N at Risk'*/
layout overlay;
entry halign=left 'N at Risk' /pad=(left=80)   textattrs=(size=8pt);
endlayout;
/*Block plots below the 'N at Risk'*/
layout overlay / walldisplay=(fill) xaxisopts=(display=none) ;
blockplot x=aval2 block=left1 /display=(values label )repeatedvalues=true
          labelattrs=(size=8pt)class=strata valueattrs=(size=8pt
          color=darkred)valuehalign=start;
blockplot x=aval2 block=left2 / display=(values label)repeatedvalues=true
          labelattrs=(size=8pt)class=strata valueattrs=(size=8pt color=darkblue)
          valuehalign=start;
endlayout;
endlayout;
```

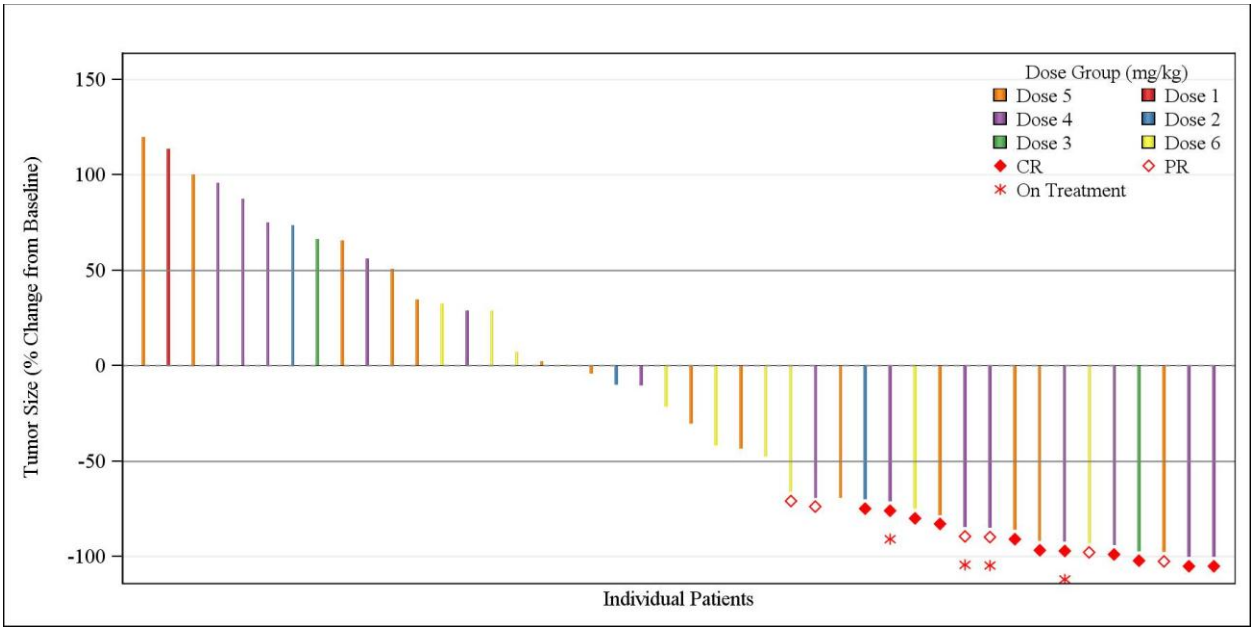
Options which are used in the above code are explained below:

- mvar – dynamic macro variables. C1ro-c5r2 is gridded within the graph. They are used for statistical interpretation.
- Layout Lattice – is used for a multi-cell grid of graphs.
- Rowweights – Represents the area for each lattice. In the above template we have 3 lattices. We are using 85% for scatter plot and step plot. 5% for 'N at Risk' and 10% for the block plot.
- Yaxisopts(linearopts) – able to define the max and min value for Y axis
- Yaxisopts(cycleattrs=true) – this specifies whether the default visual attributes of lines, markers, symbols and area fills in nested plot statements automatically change from plot to plot. When it is true, all applicable plot statements are sequentially assigned the next unused GraphDataN style statement.
- Xaxisopts(offsetmin=0.01) – Specifies the amount of offset space between the first tick and the edge of the plot. The value represents as proportion of total length of the axis. 0.01 refers as 1%
- Xaxisopts(offsetmax=0.01) – Specifies the amount of offset space between the last tick and the edge of the plot. The value represents as proportion of total length of the axis. 0.01 refers as 1%
- Xaxisopts(viewmin=0) - Represents the minimum value in the X axis
- Xaxisopts(Tickvalueslist=) – Represents the tick values on X axis
- Group – treatment groups
- Name – This is the name of the discrete legend. In this case discretelegend 'km' is created for step plot; 'cen' is created for scatter plot. The name option in the plot statement and name of the respective discretelegend has to be same.
- Discretelegend – Creates a Legend. In this case, we are using 2 separate discretelegend statements because we have stepplot and scatterplot. In the first discretelegend statement the pad bottom value specified as 30. By allowing the padded space we can prevent legends overlapping.
- Location – allows positioning the legend inside or outside the graph.
- halign – allows to move the legend position horizontally to left or right.
- Pad – Specifies a dimension to use for the extra space at left/right/top/bottom of the legend perimeter.

- Valign – allows to the legend position veritically to top or bottom.
- Valueattrs – allows controlling the font and size of the legend.
- Across=1 – specifies to place all entries in one column and start as many new rows as necessary.
- Layout gridded – Used to place the value of N, Events, Median and 95%CI inside the graph. Using the layout gridded sentence inside the layout overlay statement, allows the gridded values to be inside the graph layout.
- Blockplot – 2 separate blockplot statements to print the N at risk values 2 groups.
- Display= (values label) – This allows the values and labels to be displayed as block.
- Class – creates a separate block plot for each unique value of the specified column.
- Valuehalign – specifies the horizontal alignment of the value text within the blocks.

## Waterfall plot

Water fall plots are widely used in oncology trials. It is the traditional way of showing the tumor reduction. The following waterfall plot is used to represent the % of Tumor reduction from baseline. Each bar represents the % change in the tumor size from baseline. The bars belonging to the same dose group are represented by same color. The treatment status and objective response can be represented on the bars using special symbols.




In the above water fall plot the bars in the graph are sorted by % change in tumor from baseline. To plot the graph in this format the dataset that is passed to the sgrender procedure is sorted by descending order of % change in tumor reduction from baseline.

```
data finalgrp;
  set finalgrp(drop=x);
  by descending chpcttr;
  x+1;
run;
```

The same waterfall graph can be represented by dose group. In the graph below the bars are first sorted by dose and then by % change in tumor from baseline. In order to get this sorting order on the graph the dataset that passes to the sgrender procedure is sorted by treatment and descending order of % change in tumor reduction from baseline.

```
data finalgrp;
  set finalgrp(drop=x);
  by trtnum descending chpcttr;
  x+1;
run;
```

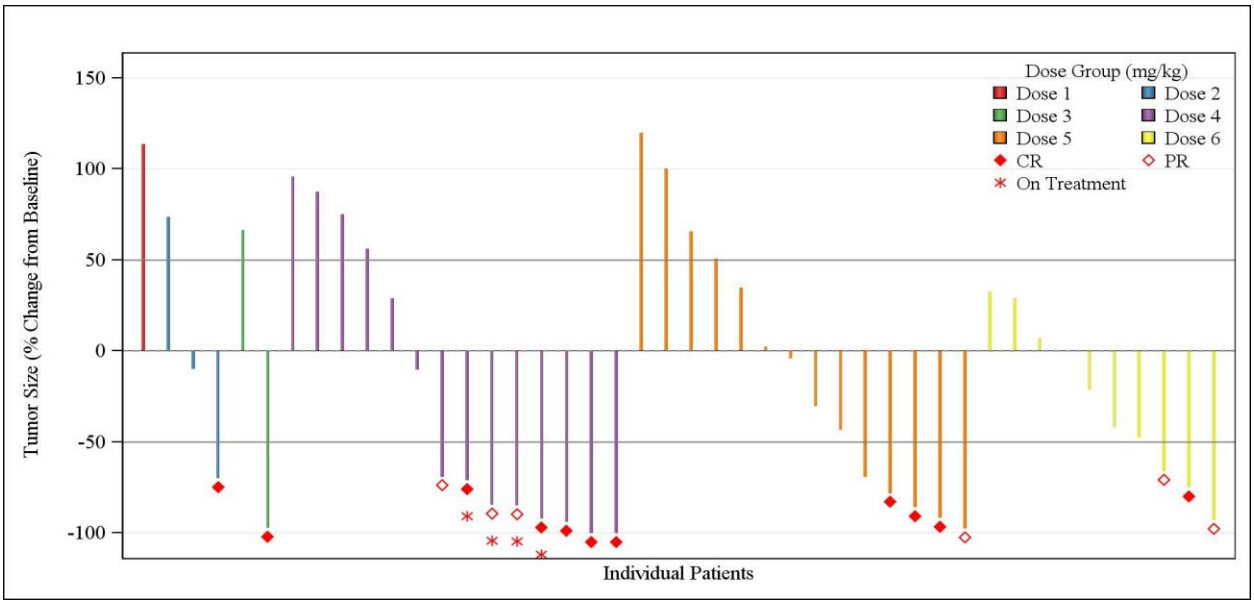



Figure 2.Waterfall Plot

Following is the template code for the water fall graph:

```
proc template;
define statgraph wf;
mvar _x _y _ysp1 _ysp2 _ysp3 _idx _grp;/*dynamic variables*/
begingraph /designwidth=9.2in designheight=4.6in;
entrytitle " ";
/*Defining X-axis and Y-axis options*/
layout overlay / yaxisopts=(type=linear linearopts=(viewmin=-100 viewmax=400
                 tickvaluelist=(-100 -50 0 50 100 150))
                 griddisplay=on label='Tumor Size (% Change from Baseline)'
                 offsetmax=0.05 offsetmin=0.05)
                 xaxisopts=(label='Individual Patients' display=(label) )
                 cycleattrs=true;
barchart y=_y x=_x / display=(fill) barwidth=0.1 index=_grp group=_grp
                     skin=modern datatransparency=0.05 name='trt';
/*Drawing reference lines*/
referenceline y=-50 / lineattrs=(pattern=1);
referenceline y=0 / lineattrs=(pattern=2);
referenceline y=50 / lineattrs=(pattern=1);

%if &cryn. %then %do;
scatterplot  x=_x y=_ysp1 /markerattrs=graphdata7   name='cr';
%end;
%if &pryn. %then %do;
scatterplot  x=_x y=_ysp2 /markerattrs=graphdata8   name='pr';
%end;
%if &ongoing. %then %do;
scatterplot  x=_x y=_ysp3 /markerattrs=graphdata9   name='ongoing' ;
%end;
```

```
/*Defining legend options*/
discretelegend 'trt'  %if &cryn.    %then %do; 'cr'      %end;
                      %if &pryn.    %then %do; 'pr'      %end;
                      %if &ongoing. %then %do; 'ongoing' %end;/
title="Dose Group (mg/kg)" titleborder=true across=2
location=inside halign=right valign=top order=rowmajor border=false;
endlayout;
endgraph;
end;
run;
```

## Duration of treatment figure:

The following graph represents the duration of treatment for each subject, on-treatment and off-treatment status and response information. The length of each bar represents the number of weeks the subject was on treatment. Different symbols and line patterns are used to represent the treatment status and response information.

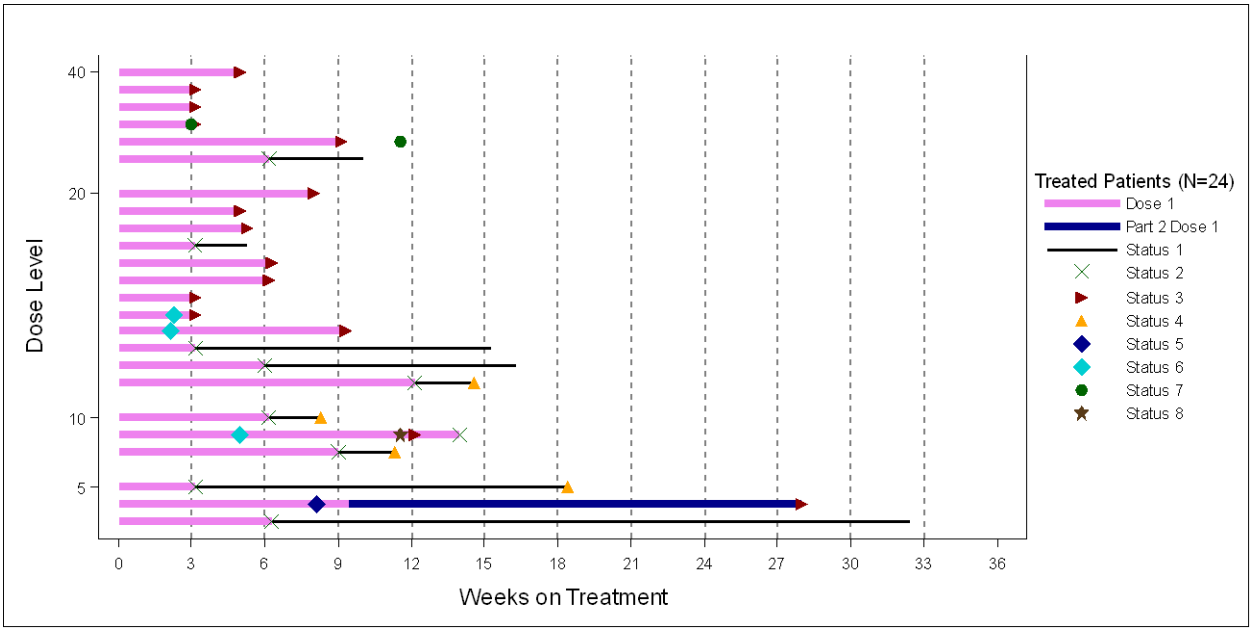



Figure 3.Duration of treatment figure

Following is the template code for Figure 3:

```
%let xmax = 36;        ** Max value (in weeks) to be displayed on the X-axis ***;
%let xview = %sysevalf(&xmax + 0.5); *** Value (in weeks) for the X-axis. ***;
%let xby = 3;                    *** Incrementing value (in weeks) for the X-
axis*;
proc template;
define statgraph vector1 ;
dynamic _y;
begingraph / designwidth=9.0in designheight=4.5in;
entrytitle " ";
layout overlay / yaxisopts=(type=linear linearopts=(viewmin=0 viewmax=&ymax
                 tickvaluelist=( 3 7 20 27))
                 label='Part A' offsetmax=0 offsetmin=0)
                xaxisopts=(type=linear linearopts=(viewmin=-0.1 viewmax=&xview
                tickvaluesequence=(start=0 end=&xmax increment=3))
                display=all label='Weeks on Treatment' ) cycleattrs=true;
%do i = &xby. %to &xmax.-&xby. %by &xby.;
referenceline x=&i / lineattrs=(pattern=2);
%end;
```

```
*** Dose 1*** ;
vectorplot  x=wk_pta xorigin=0  y=_y yorigin=_y /arrowheads=false
            lineattrs=graphdata1 (thickness=4pt) name='dose1' legendlabel="Dose 1";
*** Part 2 Dose 1*** ;
vectorplot  x=wk_ptb xorigin=wk_pta  y=_y yorigin=_y /arrowheads=false
            lineattrs=(pattern=solid thickness=4pt)group=ptbfl index=ptbfl
           name='part2dose1' legendlabel="Part 2 Dose 1";
*** Status 1*** ;
vectorplot  x=wk_fu xorigin=wk_eot  y=_y yorigin=_y /arrowheads=false
            lineattrs=graphdata6(thickness=2pt) name='Status 1'
                  legendlabel="Status 1";
*** Status 2 *** ;
scatterplot x=wk_eot y=_y /markerattrs=graphdata15 (size=12) name='status 2'
                  legendlabel="Status 2;
*** Status 3 *** ;
scatterplot x=wk_ongo y=_y /markerattrs=graphdata9 (size=10) name='status 3'
                  legendlabel="Status 3" ;
*** Status 4*** ;
scatterplot x=wk_dth y=_y /
            markerattrs=graphdata10 (size=10) name='status4' legendlabel="Status 4";
*** Status 5*** ;
scatterplot x=wk_cr y=_y /
          markerattrs=graphdata11 (size=10) name=' status5' legendlabel="Status 5";
*** Status 6*** ;
scatterplot x=wk_cri y=_y /
          markerattrs=graphdata12 (size=10) name=' status6' legendlabel=" Status 6";
*** Status 7*** ;
scatterplot x=wk_mlfs y=_y /
            markerattrs=graphdata16 (size=10) name='status7'
            legendlabel=" Status 7";
*** Status 8*** ;
scatterplot x=wk_rel y=_y /
         markerattrs=graphdata14 (size=10) name='status8' legendlabel=" Status 8";
*** Scatter Legend *** ;
discretelegend 'dose1' ' part2dose1' 'status1' ' status2' ' status3' 'Status4' '
               status5' 'status6' ' status7' ' status8' /title="Treated
         Patients (N=%cmpres(&all.))" titleattrs=(size=10pt) exclude=("No Part
         B") location=outside halign=right valign=center across=1 border=false
         displayclipped=true ;
endlayout;
endgraph;
end;
run;
```

- Tickvaluesequence - is used to add the tick values on the y axis. In the above template a list of values are given and the tick marks for the dose are drawn at the values specified in the option list.

## Mean (SD) Max % Platelets Change by Cohort

The analysis of hematology lab values is of higher importance in oncology trials. For any given lab test the mean and the standard deviation values for various dose levels can be reprasented on a graph. The following graph represents the % change in the platelets value by cohort. The diferrent dose cohorts are shown on the x-axis and the percentages on the y-axis. The standard deviation values are represented on the graph by drawing the error bars.

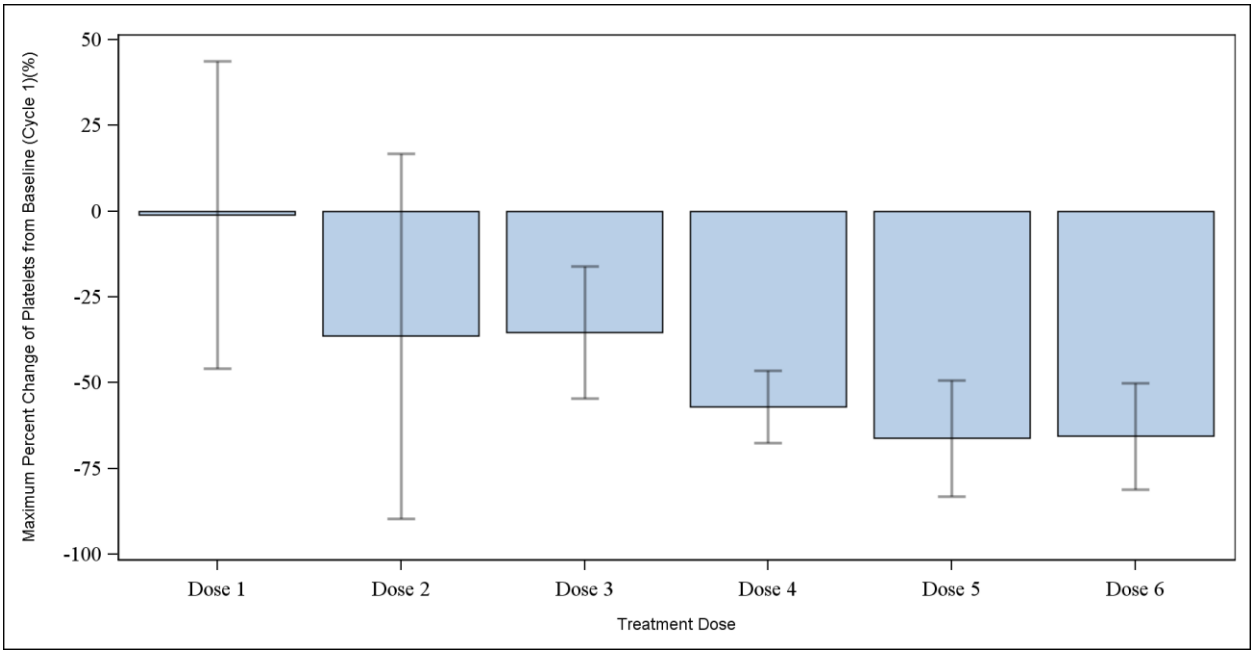



Figure 4. Mean (SD) Max % platelets change by cohort.

Following is the template code for Figure 4:

```
proc template;
define statgraph barchartparm;
begingraph / designwidth=9.2in designheight=4.75in;
entrytitle '';
/*Defining the X-axis and Y-axis options*/
layout overlay / xaxisopts=(label='Treatment Dose' )
                 yaxisopts=( label='Maximum Percent Change of Platelets from
                Baseline (Cycle 1)(%)'
                linearopts=(tickvaluesequence=(start=50 end=-100 increment=25)
                viewmin=50 viewmin=-100));
barchartparm x=arm y=mean / group=sex   name='bar' outlineattrs=(color=black);
scatterplot x=arm y=mean /  yerrorlower=l_std yerrorupper=h_std markerattrs=(size=0)
                           name='scatter' errorbarattrs=(thickness=2)/*error bars*/
                                        datatransparency=0.6;
endlayout;
endgraph;
```

- The errorbarattrs option is used to draw the error bars on the graph.

## Graph for Duration of Worst Grade Treatment-Emergent AE (SOC) by Patient

In oncology trials we often see there will be some specipfic AE's which are treatment emergent and will require some intensive analysis. The graph below represents the duration of a sepecific AE grouped by system organ class (SOC). Each patient is represented by a line on the graph and the length of the line represents the duartion. The line changes color depending up on the change in the toxicity grade. The dots in the graph represent the dosing cycles.

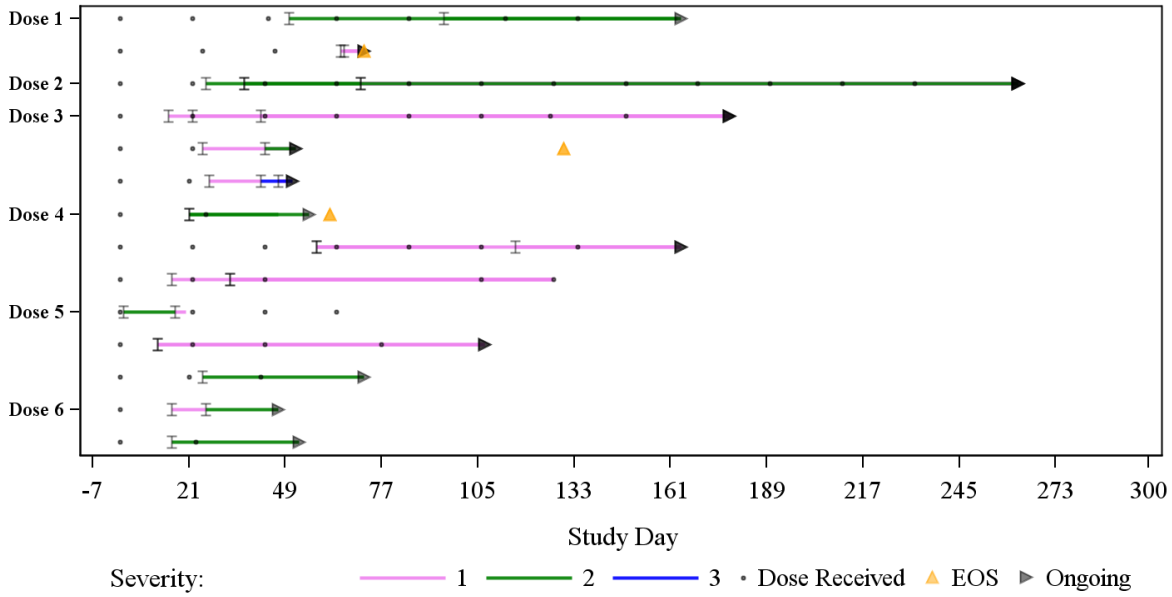



Figure 5. Duration of Worst Grade Treatment-Emergent AE (SOC) by Patient

The template code for the Figure 5 is shown below

```
proc template;
define style aerr ;
parent=rtftnr10;

    *-------------------------------------------------------------------------*
    | **Analysis Specific**
    | Colors for lines in BYGROUPS     ;
    *-------------------------------------------------------------------------*;
style GraphData1 from GraphData1 / contrastcolor=violet;
style GraphData2 from GraphData2 / contrastcolor=green;
style GraphData3 from GraphData3 / contrastcolor=blue;
style GraphData4 from GraphData4 / contrastcolor=white;
end;
run;


proc template;
define statgraph aedur;
mvar temp;
begingraph/designwidth=7.5in designheight=4.0in border=false;
entrytitle " ";
/*** plotting the dose on the y-axis***/
layout  overlay/ yaxisopts=(linearopts=(tickvaluelist=(2 5 8 11 12 14 )
                tickvalueformat=revdose.) tickvalueattrs=(size=8 weight=bold)
                label=" ")
                xaxisopts=(label="Study Day"
                linearopts=(tickvaluesequence=(start=-7 end=300 increment=14)
                viewmin=-7.0 viewmax=300.0));
vectorplot x=aeendy y=subj_ae_id xorigin=aestdy yorigin=subj_ae_id/
                arrowheads=false group=aetoxgr index=index1 name='sev'
                legendlabel="Severity" lineattrs=(thickness=2 pattern=solid)
                datatransparency=0.1;
```

```
/*** for plotting the dosing cycles ****/
scatterplot x=xdose y=ydose/
               markerattrs=(symbol=circlefilled size=3) name='dos'
               legendlabel="Dose Received" datatransparency=0.5;
scatterplot x=xstmark y=subj_ae_id/
               markerattrs=(symbol=ibeam)datatransparency=0.5;

/**** to display the EOS ***/
scatterplot x=ongoing y=subj_ae_id/markerattrs=(symbol=trianglerightfilled
                color=black) name='ongo' legendlabel="Ongoing"
                datatransparency=0.5 ;
scatterplot x=eosdy y=subj_ae_id/markerattrs=(symbol=trianglefilled
color=orange)                name='eos' legendlabel="EOS" datatransparency=0.5;

/** to display the legend ***/
discretelegend 'sev' 'dos' 'ongo' 'eos' /title="Severity: " border=false
                sortorder=ascendingformatted;
endlayout;
endgraph;
end;
run;
```

## CONCLUSION

This paper gives a basic understanding of ODS template and how the options are selected when used with Graph template. These graph templates can be used as a base which further can be used to build different kind of graphs. GTL has numerous options which help the graph creation easier and efficient. GTL has also allowed a reduction in the quantity of annotations which are used in ODS graphics.

## REFERENCES


1. Schwartz, Susan. “Clinical Trial Reporting Using SAS/GRAPH® SG Procedures.” *Proceedings of the SAS Global 2009 Conference*.
2. Matange, Sanjay. “Clinical Graphs using ODS Graphics.”
3. Larsen, Jan Rene. “Clinical Graphs using ODS Graphics – Analysis of Safety Data for Clinical Trials.”
4. Manish Kumar Goel, Pardeep Khanna, and Jugal Kishore. “Understanding survival analysis: Kaplan-Meier estimate”.


## ACKNOWLEDGMENTS

We would like to thank Gordon Fancher, Manager at Seattle Genetics Inc. for his technical guidance on GTL.

## CONTACT INFORMATION

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


Name: Avani Kaja
Enterprise: Seattle Genetics, Inc.
Address: 21823 30th Drive Southeast
City, State ZIP: Bothell, WA 98021
E-mail: akaja@seagen.com

Name: Murali Kanakenahalli
Enterprise: Seattle Genetics, Inc.
Address: 21823 30th Drive Southeast
City, State ZIP: Bothell, WA 98021
E-mail: mkanakenahalli@seagen.com