ABSTRACT
Graphical representation of clinical data is used for concise visual presentations of the results. SAS® programmers are frequently asked to generate Kaplan-Meier plots, along with graphs of ECG, and clinical laboratory data. Although Kaplan-Meier plots are easily generated using PROC LIFETEST, reviewers often require additional information such as the number of patients at risk. The Annotate facility in SAS/GRAPH® allows us to create customized modification to the graphics output. In this paper I will describe how to create annotations in Kaplan-Meier and line graphs from statistical procedures. Furthermore, these methods can be applied to annotations of other types of graphs.

INTRODUCTION
The Annotate facility in SAS/GRAPH® is a very useful tool in enhancing or changing feature of your graphical output. Annotate gives you the flexibility to add any drawing object or text, in any format or color, to any place on your graphical output. With the help of examples I will explain how to generate Annotate datasets and produce customized graphics output.

SURVIVAL ANALYSIS PLOTS
Kaplan-Meier survival estimates created during time-to-event analyses are commonly used in clinical trials. The graphic presentation of survival analysis is a significant tool which facilitates clear understanding of the underlying events.

Consider the following example dataset with subject identifier (SUBJID), time to progression in weeks (TIME_P, progressive disease (PD) and treatment group (Group) variables. The quick survival plot shown in figure 1 can be created and saved in an RTF file using the ODS RTF destination and the following Proc Lifetest code

```sas
ODS RTF FILE='location\Pharmasuggraph.rtf' style=statistical;
ODS LISTING CLOSE;
ODS GRAPHICS on;
Proc Lifetest data=all outsurv=ind1 method=km;
  Time time_p * pd (0);
Run;
ODS GRAPHICS OFF;
ODS RTF CLOSE;
ODS LISTING;
QUIT;
```

The ODS GRAPHICS ON statement will add statistical graphics to the output. Although plots are easily generated through PROC LIFETEST reviewers often require additional information such as number of patients at risk or log rank statistics to better interpret the Kaplan-Meier plots. The SAS Annotate facility allows us to create customized modifications to the graphics output. The Annotate facility is included within SAS/GRAPH® and acts as a bridge between the procedure selected by the user and the user's desire to customize the graphics output.

ANNOTATE DATASET VARIABLES

The first step is to create the Annotate dataset in order to use the Annotate facility of the SAS/GRAPH®. The Annotate dataset is then fed to a SAS/GRAPH procedure and gets processed along with the SAS/GRAPH code. An Annotate dataset is similar to other SAS datasets but contains variables with predefined names. The Annotate variables tell Annotate what to do, how to do it, or where to do it. Below is a table defining some of the important variables that will be used in my examples. For a complete list of Annotate variables, refer to the SAS/GRAPH documentation.

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>FUNCTION</td>
<td>Specifies the Annotate drawing action. Table 2 below gives a list of important functions.</td>
</tr>
<tr>
<td>X</td>
<td>The numeric horizontal coordinate</td>
</tr>
<tr>
<td>Y</td>
<td>The numeric vertical coordinate.</td>
</tr>
<tr>
<td>HSYS</td>
<td>The type of units for the size (height) variable.</td>
</tr>
<tr>
<td>XSYS</td>
<td>The coordinate system for the X variable.</td>
</tr>
<tr>
<td>YSYS</td>
<td>The coordinate system for the Y variable.</td>
</tr>
<tr>
<td>LINE</td>
<td>Line type of graphics item.</td>
</tr>
<tr>
<td>POSITION</td>
<td>Placement/alignment of text.</td>
</tr>
<tr>
<td>COLOR</td>
<td>Color of graphics item</td>
</tr>
<tr>
<td>SIZE</td>
<td>Size of the graphics item. Specific to the function. For example, size is the height of the character for a label function.</td>
</tr>
<tr>
<td>STYLE</td>
<td>Font/pattern of a graphics item.</td>
</tr>
<tr>
<td>TEXT</td>
<td>Text to use in a label, symbol, or comment.</td>
</tr>
</tbody>
</table>
FUNCTIONS
The Annotate dataset FUNCTION variables tell SAS what to do. The other variables all modify, specify or locate the function. Below is a table listing some of the functions that will be used in our example. For a complete list of Annotate functions, refer to the SAS/GRAPH documentation.

TABLE 2. Functions

<table>
<thead>
<tr>
<th>FUNCTION</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>LABEL</td>
<td>Draws text.</td>
</tr>
<tr>
<td>MOVE</td>
<td>Moves to a specific point.</td>
</tr>
<tr>
<td>DRAW</td>
<td>Draws a line from the current position to a specified position.</td>
</tr>
<tr>
<td>COMMENT</td>
<td>As a documentation aid, allows you to insert a comment into the SAS Annotate file.</td>
</tr>
<tr>
<td>BAR</td>
<td>Draws a rectangle from the current position to a specified position.</td>
</tr>
<tr>
<td>SYMBOL</td>
<td>Draws a symbol</td>
</tr>
</tbody>
</table>

COORDINATE SYSTEMS
The Annotate facility recognizes three different drawing areas, each with two possible unit types and the ability to designate these unit types as relative or absolute. These 12 conditions are coded as 1-9, A, B and C based on Figure 1 below.

FIGURE 2. Areas and their coordinate systems. [1]

The Data Area represents only the space within the graph axes. The second graph in the middle of figure 2 shows the Graphics Output Area which is the entire writable page of output. The bottom graph shows the Procedure Output Area or the area taken up by the graphic object. Once you have determined the appropriate coordinate system for each of your dimensions, you put the coded value in the XSYS and YSYS variables.
CREATING ANNOTATE DATASET

The first step is to create an Annotate dataset for the labels in the KM graph. Consider the following code.

``` SAS
DATA ANNO1;
  length function $8 text $100;
  retain xsys ysys '4' hsys '9' function 'label';  
  line=1;
  size=2.5;
  position='6';
  x=1;  y=4.0;
  text="Number of Subjects at Risk";
  output;
  x=1;y=2.8;
  text="Grp1";
  output;
run;
```

1. The coordinate system selected for the x axis and y axis is '4'. This corresponds (see Figure 1) to the Graphics output Area with absolute values. With this coordinate system you can use the actual values of the point of interest to place the label. Multiple variable assignments can be made in a single statement and multiple RETAIN statements are also permitted.

2. The completed observation is written to the Annotate dataset through the use of the output statement. Each statement provides a value for an Annotate variable. After you have assigned all of the variables values for an observation, you must use an OUTPUT statement to write the observation to the dataset.

The second step is to create the Annotate dataset for the number of subjects at risk. In order to get the number of subject at risk we need to do some data manipulation before creating the Annotate dataset.

```
***************
To obtain Number of Subjects at Risk
***************;
*Step 1:  Create frame dataset with all possible weeks within each group;
*&xmax1 contains the maximum value of time_p;
*&maxcol contains maximum treatment group. In our example it is 1;
```

``` SAS
data frame;
  do time_p = 0 to &xmax1;
    do group=1 to &maxcol;
      output;
    end;
  end;
run;
```

``` SAS
proc sort data=frame;
  by group descending time_p;
run;
```

*** Step 2: Get subjects counts from SURVIVAL data at each time point;
```
proc sort data=survival ;
  by group descending time_p pd;
run;
```

``` SAS
data count1;
  set survival;
  by group descending time_p pd;
  count+1;
  if first.group then count=1;
  if last.time_p then n=count;
run;
```

``` SAS
proc sort data=count1 out=count2;
  by group time_p n;
  where n;
run;
```
**proc sort data=count2(keep=group time_p n);**
  by group descending time_p ;
run;

*** Merge frame and count2 dataset to get count at each time point;***
*** Retain value for previous missing values;***
Data count;
  merge frame count2;
  by group descending time_p;
  if first.group then last=0;
  if n=. then n=last;
  last=n;
  retain last;
run;

The COUNT dataset contains the number of subjects at each time point. Next the Annotate dataset is created using the following code.

data anno2;
  length text $200 function $8;
  set %do i = 0 %to &xmax1 %by 10;
    COUNT(where=(time_p=&i)) %end;;
  retain y ;
  function='label';
  xsys='2';  ysys='4';
  position='5';
  size=1.0;
  if n ne . then text=trim(left(put(n,4.)));
  x=time_p;
  y=1.6;
run.;

Select the number of subjects at risk at every 10 weeks.

We concatenate the two datasets to get the final Annotate dataset as

data anno;
  set anno1 anno2;
run;

This will be our final Annotate dataset. We can now easily generate the graph using this Annotate dataset. PROC GPLOT is used to create the final plot shown in figure 3.

GOPTIONS DEVICE=JPEG TARGETDEVICE =JPEG HSIZE=6 IN VSIZE=5 IN
  GSFILENAME=graphs GSFMODE=REPLACE FTEXT="ARIAL" noborder ;

title1 h=12pt j=c "Time to Progression Plot";
symbol1 line=1 color= blue width=1 interpol=stepjl v=none;
symbol2 line=1 color= black width=1 interpol=none v=circle height=.8 ;
axis1 order = (0 to &xmax1 by 10)
  label = (h = 1.2 "Time to Progression (weeks)"
            minor = (number = 3)
            offset = (0,0));
axis2 order=(0 to 1 by .2)
  label = (h = 1.2 angle = 90 "Survival Probability"
            minor = (number = 3)
            offset = (0,0));
legend1 label=('')
  mode = share
  order = (1)
  position=(TOP RIGHT INSIDE)
  value=("Censored values")
  down = 2
  frame;
Notice the addition of the option
Annotate=anno;

to the plot statement. This allows the plot to incorporate the Annotate instructions found in the Annotate dataset. When a
procedure uses an Annotate dataset it reads and interprets the observations one at a time, starting with the first observation
and proceeding to the last. The order of the observations in the dataset determines the order in which the graphics elements
are generated. If the coordinates of two graphics element overlap, the graphics element produced by an earlier observation
can be overwritten by any graphics elements that are produced by subsequent observations.

FIGURE 3. Annotated Kaplan-Meier Survival Estimate Plot.

ECG GRAPHS

Changes in ECG parameters like QRS, QT, and QTc are summarized over time in clinical trials. ECG graphs are generated to
display 95% CI interval for Mean Change in ECG parameter and also number of subjects at risk. In the following I will explain
how to annotate the line graph to display additional information.

For the following example we have used a dataset with variables subject identifier (SUBJID), visit (VISITP), treatment group
(TRT1PN) and change from baseline (CHG).
**Compute CI for change from baseline;**

```sas
proc summary nway alpha=0.05 data=all;
   class trt1pn visitp;
   var chg;
   where visitp > 1;
   output out=stats mean(chg)=meanchg n = x lclm(chg)=lclmchg uclm(chg)=uclmchg;
run;
```

The confidence intervals and mean change from baseline are derived using PROC SUMMARY. The following Annotate dataset is created to annotate the ECG graph.

**Create Annotate dataset to draw labels;**

```sas
data annol;
   length function $ 8 text $ 50;
   retain xsys ysys '4' function='label';
   position='6';
   size=1;
   x=1; y=6.6;
   text="No. of Subjects with ECG Parameter";
   output;
   x=1; y=5.4;
   text="All subjects";
   output;
   y=3.0;
   text="Group 1";
   output;
run;
```

**Create Annotate dataset to draw vertical lines and symbols for CIs;**

```sas
data anno2;
   length color function text $ 8;
   set stats;
   if trt1pn=1 then text = 'dot';
   else if trt1pn=2 then text = 'circle';
   else text = 'diamond';
   if lclmchg > .Z and uclmchg > .Z then do;
      function='move'; xsys='2'; ysys='2'; x=visitp; y=meanchg; output;
      function='draw'; xsys='2'; ysys='2';
      x=visitp; y=uclmchg; line=1; size=1; output;
      function='symbol'; xsys='2'; ysys='2';
      x=visitp; y=uclmchg; line=1; size=.8; output;
      function='move'; xsys='2'; ysys='2';
      x=visitp; y=meanchg; color=''; line=.; size=.; output;
      function='draw'; xsys='2'; ysys='2';
      x=visitp; y=lclmchg; line=1; size=1; output;
      function='symbol'; xsys='2'; ysys='2';
      x=visitp; y=lclmchg; line=1; size=.8; output;
   end;
run;
```

The coordinate system selected for the x axis and y axis is ‘2’. This corresponds (see Figure 1) to the Data Area with absolute values.

**Create the number of subjects with lab values over time on the x-axis per group**

```sas
data frame1;
   do trt1pn=1, 3;
      do visitp = 0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 26;
         count = 0;
         output;
      end;
   end;
   format trt1pn trtft.;
run;
```
**proc** sort **data=all**;
   **by** trt1pn visitp;
**run**;

**data** time_p_n (keep=trt1pn visitp count);
   **set** ecgdataset;
   **by** trt1pn visitp;
   if first.visitp then count=0;
   count+1;
   if last.visitp and count > . then output;
**run**;

*** Retain value for previous missing values ;
**data** time_p_n;
   **merge** frame1 (in=inf) time_p_n1;
   by trt1pn visitp;
   if inf;
**run**;

**data** anno3;
   length text $200 function $8;
   **set** time_p_n (**where**=(trt1pn > .Z));
   retain y xsys '2' ysys '4' function='label';
   position='5';
   size=1.0;
   if count > .Z then text=trim(put(count, 2.));
   x=visitp;
   if trt1pn=1 then y = 4.2;
   else if trt1pn=3 then y = 1.8;
**run**;

**data** anno;
   **set** anno1 anno2 anno3;
**run**;

Anno is the final Annotate dataset. PROC GPLOT is used to create the final plot shown in figure 4. Axis1, Axis2 and Legend1 are defined in the program but not shown here.

**proc** gplot **data =** stats **annotate=anno**;
   **plot** meanchg * visitp = trt1pn/
      haxis = axis1
      vaxis = axis2
      legend = legend1
      href = 24 lhref=3 /* dotted vertical reference line at Month 24 */
      noframe;
**run**;
**quit**;
CONCLUSION
Annotate is a powerful, flexible tool that allows you to draw any text or object on any graphical output. This paper has explained some of the Annotate facility with the help of examples in clinical trials. Applying this to selected situations enables the reviewers of the graphs to gather more precision around the visual representation of the results.

REFERENCES

ACKNOWLEDGMENTS
I would like to thank Nancy Brucken, Mark Matthews, Gokul Shankar and Steven Hege for guiding this work and carefully reviewing the paper with comments and suggestions.
CONTACT INFORMATION

Your comments and questions are valued and appreciated. The author can be reached at:

Sudhir Singh
i3 Statprobe
1001 Winstead Drive, Suite 200
Cary, NC 27513 U.S.A
Email: sudhir.singh2@i3statprobe.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.