Using the Annotate Facility for Enhancing Output
Marcia Branham and Timothy Maxwell, STATPROBE, Inc. Lexington KY

ABSTRACT
Table presentation is an important consideration when programming results for clinical trials. Attention and readability are greatly increased when using annotate to program tables. The flexibility for output is virtually endless. A graphical display of serious adverse event outcomes can be shown using all outcomes. Additionally, a piece of pie can be exploded to emphasize the most severe outcome. Furthermore, a table to the side of the pie could show the characteristics of these patients. The graphical presentation of this data is much more readily understood and comprehended. The annotate facility can accommodate a multitude of useful tools, both simple and very ornate, to enhance graphical display.

Simple concepts will be discussed to get you started with the annotate facility.

INTRODUCTION
Annotate is a facility within SAS/GRAPH® that gives the programmer ultimate control over what you see on a graph. The ideal place to start when creating a graphic is to sketch ideas for the appearance of the end result. A useful tool for this can be graph paper or your word processor. As we will see later, the grids on graph paper can be useful as SAS/GRAPH uses a grid system to layout where things are positioned. Many times you’ll have several ideas; draw them all, you may find the best result will be a combination of two or three ideas.

After the basic graphics have actually been programmed and produced, you may find several components need to be changed to make the graph more appealing. This may include placement, size, fonts, shading, shading patterns and titles, just to name a few.

You must know ahead of time what data set(s) you’re working with and what statistics are to be displayed on the graphic. In this example, there are two SAS® data sets to work with to create the graphic. The first contains only serious adverse event outcomes; the second contains demographic information for all patients. All outcomes of serious adverse events will be displayed graphically; demographic information for only the most serious outcome will be identified and be displayed in a table to the side of the pie.

Six outcomes of adverse events have been identified as being serious, with death being the most serious. Proc SQL will first be used to obtain the N’s and percents for each ‘slice of the pie’. Table 1 below is the output for each ‘slice’:

<table>
<thead>
<tr>
<th>SAE</th>
<th>SAE_N</th>
<th>SAE_P</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>98</td>
<td>17.9159</td>
</tr>
<tr>
<td>2</td>
<td>52</td>
<td>9.5064</td>
</tr>
<tr>
<td>3</td>
<td>154</td>
<td>28.1536</td>
</tr>
<tr>
<td>4</td>
<td>64</td>
<td>11.7002</td>
</tr>
<tr>
<td>5</td>
<td>132</td>
<td>24.1316</td>
</tr>
<tr>
<td>6</td>
<td>47</td>
<td>8.5923</td>
</tr>
</tbody>
</table>

Patients who died (SAE = 6) is the most serious event, so final, yet very understated in the above table. As programmers, our job is often to tell a story. When we complete this graph, we will have a picture of related numbers and characters, when put all together, make sense and will have a much greater impact than the table above.

IDENTIFYING AND JOINING DEMOGRAPHICS WITH THE MOST SERIOUS OUTCOME

The variable called SAE contains all outcomes of serious adverse events with 6 being death. We will first identify the demographics of these patients and calculate the denominator used to calculate the individual percentages for each ‘slice of the pie’. Table 1 below is the output for each ‘slice’:

Our first SAS data set contains only serious adverse event outcomes. In order to get the N’s and percents, we start with SAS code as follows:

```
Proc sql;
   Create table denom as
      Select count(*) as denom
      From sug.demog
   ;
   Create table saecount as
      Select distinct sae, count(sae) as sae_n, count(sae)/denom*100 as sae_p
      From sug.demog, denom
      Group by sae
   ;
```

This counts all outcomes of serious adverse events to obtain the denominator used to calculate the individual percentages for each ‘slice of the pie’. Table 1 below is the output for each ‘slice’:

<table>
<thead>
<tr>
<th>SAE</th>
<th>SAE_N</th>
<th>SAE_P</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>98</td>
<td>17.9159</td>
</tr>
<tr>
<td>2</td>
<td>52</td>
<td>9.5064</td>
</tr>
<tr>
<td>3</td>
<td>154</td>
<td>28.1536</td>
</tr>
<tr>
<td>4</td>
<td>64</td>
<td>11.7002</td>
</tr>
<tr>
<td>5</td>
<td>132</td>
<td>24.1316</td>
</tr>
<tr>
<td>6</td>
<td>47</td>
<td>8.5923</td>
</tr>
</tbody>
</table>

Please note that all data we are using is totally fake.

Before we plunge into the annotation, we will review each step of the process as follows:

- Identify outcomes of serious adverse events
- Identify and join demographics with the most serious outcome
- Finalize the data to create the pie for the outcomes of serious adverse events
- Placement of titles and legends on the graph
- Create the pie of outcomes of serious adverse events
- Create the demographic table for the most serious outcome

IDENTIFYING OUTCOMES OF SERIOUS ADVERSE EVENTS

Create table sexdenom as
Select sex, count(sex) as sex_n
From sug.demog(where=(sae=6))
Group by sex
;
Create table demogcnt as
Select distinct a.sex, count(age) as over75_n, count(age)/sex_n*100 as over75_p,
Continue Code. . .
From sug.demog(where=(sae=6))
  Group by sex
;
Create table demogcnt as
  Select distinct a.sex,
  count(age) as over75_n,
  count(age)/sex_n*100 as over75_p,
  count(smoke) as smoke_n,
  count(smoke)/sex_n*100 as smoke_p,
Continue Code. . .
From (Select sex, case
  when age=1 then 1
  else .
end as age,
  case
  when smoke=1 then 1
  else .
end as smoke,
Continue Code. . .
From sug.demog(where=(sae=6))) as a
left join
  Sexdenom as b
On a.sex=b.sex
Group by a.sex
;
Quit;
Data demogcnt (drop=sex_n);
Merge demogcnt sexdenom;
By sex;
Totsex = 'n='||compress(put(sex_n,8.));
Run;

We have calculated denominators for all patients and denominators by
sex for deceased patients. We then pull out only those patients from
the demographic data set and perform calculations to be placed later in
the table. Table 2 below is partial output as follows:

Table 2

<p>| | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>V</td>
<td>V</td>
<td>R</td>
<td>R</td>
<td>R</td>
<td>R</td>
<td>R</td>
</tr>
<tr>
<td>T</td>
<td>E</td>
<td>E</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>A</td>
</tr>
<tr>
<td>O</td>
<td>R</td>
<td>R</td>
<td>C</td>
<td>C</td>
<td>C</td>
<td>C</td>
</tr>
<tr>
<td>T</td>
<td>7</td>
<td>7</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
</tr>
<tr>
<td>S</td>
<td>5</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
</tr>
<tr>
<td>X</td>
<td>X</td>
<td>N</td>
<td>P</td>
<td>N</td>
<td>P</td>
<td>P</td>
</tr>
<tr>
<td>1</td>
<td>n=29</td>
<td>15</td>
<td>51.7241</td>
<td>10</td>
<td>34.4828</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>n=18</td>
<td>10</td>
<td>55.5556</td>
<td>8</td>
<td>44.4444</td>
<td>5</td>
</tr>
</tbody>
</table>

FINALIZING THE DATA TO CREATE THE PIE FOR
THE OUTCOMES OF SERIOUS ADVERSE EVENTS

First we transpose the output from the serious adverse event outcomes output (re: Table 1) and get the total denominator to later display at the bottom of the pie. The code is as follows:

Proc transpose data=saecount
out=saecount prefix=sae;
  Var sae_p;
  Id sae;
Run;

Data saecount (drop=denom);
  Merge saecount denom;
  Totsae = 'N='||compress(put(denom,8.));
Run;

This gives us numeric percentage values only used in annotate to determine the sizing of each 'slice of pie'. Table 3 below shows the output from the transposition of the database:

Table 3

<table>
<thead>
<tr>
<th>SAE1</th>
<th>SAE2</th>
<th>SAE3</th>
<th>SAE4</th>
<th>SAE5</th>
<th>SAE6</th>
<th>TOTSAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>17.91</td>
<td>9.50</td>
<td>28.15</td>
<td>11.70</td>
<td>24.13</td>
<td>8.59</td>
<td>N=547</td>
</tr>
</tbody>
</table>

Before we start the graph, we will do one last data step. This will set the pie and table data together. The first record will contain the pie data (SAE’s); the 2nd and 3rd record will contain the table data (demographics).

Data saecount;
  Set saecount Demogcnt;
Run;

TITLES AND LEGENDS FOR THE GRAPH

We start with some basic instruction to the annotate macros within SAS/GRAPH software.

The syntax as defined below is not meant to replace any syntax as defined by the SAS Institute.

The syntax of the annotate macros we’ll be using in this section (and throughout) are as follows:

%Label(x, y, text, color, angle, rotate, height, font, position);
  - X and Y are the coordinates of placement on the page.
  - Text can either be written text or a variable.
  - Color can be anything from black to shades of gray to red. We’ll get more into color explanation further down.
  - Angle rotates the entire text, an example of this would be placing text horizontally on the X-axis of a graph.
  - Rotate rotates individual characters.
  - Height is the size of the font.
  - There are many fonts, what you can actually use is dependent on the type of printer you’ll be using.
  - Position is the label placement; 4=right, 5=left, 6=center are the ones we mainly use although there are several others.

%Line(X1, Y1, X2, Y2, color, type, width);
  - X1 and Y1 are the coordinates as to where the line begins.
  - X2 and Y2 are the coordinates as to where the line ends.
  - Type can be solid (1), broken (33), among many others.
  - Width can also be defined as thickness, typically will use a 1; the higher the number the thicker the line; the restriction is the line can only be the size of the Y coordinate. For a very bold line you’d need to use %Bar.

%Rect(X1, Y1, X2, Y2, color, type, width);
  - See %Line above for explanations. The only difference is you’ll be placing X and Y coordinates at opposite corners to draw a rectangle instead of a line.

%Label(x, y, text, color, angle, rotate, height, font, position);
  - X and Y are the coordinates of placement on the page.
  - Text can either be written text or a variable.
  - Color can be anything from black to shades of gray to red. We’ll get more into color explanation further down.
  - Angle rotates the entire text, an example of this would be placing text horizontally on the X-axis of a graph.
  - Rotate rotates individual characters.
  - Height is the size of the font.
  - There are many fonts, what you can actually use is dependent on the type of printer you’ll be using.
  - Position is the label placement; 4=right, 5=left, 6=center are the ones we mainly use although there are several others.

%Line(X1, Y1, X2, Y2, color, type, width);
  - X1 and Y1 are the coordinates as to where the line begins.
  - X2 and Y2 are the coordinates as to where the line ends.
  - Type can be solid (1), broken (33), among many others.
  - Width can also be defined as thickness, typically will use a 1; the higher the number the thicker the line; the restriction is the line can only be the size of the Y coordinate. For a very bold line you’d need to use %Bar.

%Rect(X1, Y1, X2, Y2, color, type, width);
  - See %Line above for explanations. The only difference is you’ll be placing X and Y coordinates at opposite corners to draw a rectangle instead of a line.
%Bar(X1, Y1, X2, Y2, color, type, pattern);
- The X and Y coordinates will be set up like a %Rect coordinrats depend on the sizing).
- Type can be draw edges (0), adjust vertically (1), adjust horizontally (2), no edges drawn (3).
- Patterns include SOLID, R1 (slanted lines), and many others.

For a landscape graph, on 8 ½ x 11 paper, the X coordinates are 0 to 70 (VPOS) and they Y coordinates are 0 to 170 (HPOS). VPOS and HPOS are defined in the GOPTIONS statement.

Now we begin the graph! The first part compiles the annotate macros (%Annomac), defines the graphic options (Goptions), followed by titles, framing the page, and placement of the legends. The part of this SAS code is as follows:

```
Options ls=155;

%Annomac;

Goptions reset=goptions
device=lxps4079 noborder noprompt
rotate=landscape vsize=7.5 in
hsize=10.0 in vorigin=0.25 in
horigin=0.5 in vpos=70 hpos=170
nodeisplay;

Data output;
  Retain txtrot;
  Set saecount;
%Dclanno; *Initialize annotate variables;
  length text $200;

****** TITLES *****;
%Line(2,66.5,169,66.5,black,1,1);
%Rect(2,3,169,61,black,1,2);
%Label(2, 67.5, "XYZ Pharmaceutical Company", black, ., ., 1.5,
hwps1009, 6);
  %Label(169, 67.5, "Page 1", black, ., ., 1.5, hwps1027, 4);
%Label(2, 65.5, "May 2000", black, ., ., 1.5, hwps1027, 6);

%Label(2, 63, "Outcomes of Serious Adverse Events (SAE’s) for All Patients", black, ., ., 2, hwps1027, 6);
*SUPERSCRIPT FOR FOOTNOTE, BESIDE THE TITLE;
  %Label(88.5, 63.5, "1", black, ., ., 1, hwps1027, 6);

%Label(40, 56, "All SAE Outcomes", black, ., ., 2, hwps1027, 5);

*FOOTNOTES;
  %Label(2, 2, '1', black, ., , 0.8, hwps1027, 6); * Superscript;
    %Label(3, 1.5, "All Data Contained in this Graph is totally fake.", black, ., ., 1, hwps1009, 6);

After the titles are set up, we will add the legend for the pie. Up until now the only color we’ve used is black. Now we will move on to colors such as red (used for ‘Death’), or a combination of hue and prefix for a color; in the case of BIB (used for ‘Continuing’), this stands for ‘Bright Blue’. Colors such as green, yellow, blue, and different shades of gray can also be used. What can be used is dependent on the printer device being used.

****** LEGEND *****;
%Rect(8, 5.5, 162, 12.5, black, 1, 1);
%Rect(7.25, 5, 162.75, 13, black, 1, 2);
%Bar(12, 9.5, 16, 11.5, red, 3, solid);
%Rect(12, 9.5, 16, 11.5, black, 1, 1);
%Label(17.5, 10.5, "Death", black, ., ., 1.25, hwps1027, 6);
%Bar(12, 6.5, 16, 8.5, BIB, 3, solid);
%Rect(12, 6.5, 16, 8.5, black, 1, 2);
%Label(17.5, 7.5, "Continuing", black, ., ., 1.25, hwps1027, 6);
%Bar(70, 9.5, 74, 11.5, LIG, 3, solid);
%Rect(70, 9.5, 74, 11.5, black, 1, 1);
%Label(75.5, 10.5, "Extended Hospital Stay", black, ., ., 1.25, hwps1027, 6);

Continue Code. . .
CREATING THE PIE

We'll be introducing 3 additional annotate macros for the pie. The syntax of these are as follows:

%Slice(x, y, angle, rotation, size, color, style, line);
- We use BOTH for the value of LINE. This tells the arc to draw both leading and trailing lines. Other valid values are no radius lines (WHOLE or NONE), lead point to center line only (LEAD), trail point to center line only (TRAIL).

%Piexy(angle, multiplier);
- This assigns the last coordinate to the text coordinate (notice when drawing the pie, when we invoke the %LABEL macro, both the X and Y coordinates are missing; this macro assigns these).

*** CODE FOR THE PIE ***;

If _N_ = 1 then do;
  txtrot = 0;
  %Slice(50, 37, 3.6*(0-sae6/2), 3.6*(sae6), 15, red, solid, both);
  %Piexy(txtrot, .7);
  %Cntl2txt;
  %Label(., ., compress(put(sae6, 8.1)||'%'), black, ., ., 2, hwpsl027, 5);
  txtrot = txtrot + .5*3.6*(sae2 + sae3);

  %Slice(40, 37, ., 3.6*sae1, 15, BIY, solid, both);
  %Piexy(txtrot, .7);
  %Cntl2txt;
  %Label(., ., compress(put(sae3, 8.1)||'%'), black, ., ., 2, hwpsl027, 5);
  txtrot = txtrot + .5*3.6*(sae3+sae4);

  %Slice(40, 37, ., 3.6*sae4, 15, VILG, solid, both);
  %Piexy(txtrot, .7);
  %Cntl2txt;
  %Label(., ., compress(put(sae5, 8.1)||'%'), black, ., ., 2, hwpsl027, 5);
  txtrot = txtrot + .5*3.6*(sae5);

  %Slice(50, 37, 3.6*(0-sae6/2), 3.6*(sae6), 15, black, empty, both);
  %Slice(40, 37, ., 3.6*sae1, 15, black, empty, both);
  %Slice(40, 37, ., 3.6*sae2, 15, black, empty, both);
  %Slice(40, 37, ., 3.6*sae3, 15, black, empty, both);
  %Slice(40, 37, ., 3.6*sae4, 15, black, empty, both);
  %Slice(40, 37, ., 3.6*sae5, 15, black, empty, both);
  %Slice(40, 37, ., 3.6*0-sae6, 15, black, empty, both);

  %Label(40, 18, totsae, black,...,2,hwpsl027,5);
End;
With the pie drawn on the graph, it now appears as follows:

**CREATING THE DEMOGRAPHIC TABLE**

We'll be introducing 2 additional annotate macros for the table. The syntax of these are as follows:

- `%Poly(x, y, color, style, line);`
  - This simply is specifying the X and Y coordinates as to where the polygon will start in addition to color, etc.

- `%Polycont(X,Y);`
  - This continues drawing the polygon. The color, style and line are carried over from the `%POLY` macro.

The first section creates the box for the table. To draw the box with a 3-D effect, the first 2 parts are using a lighter color of red (VIPK=vivid pink passed in `%POLY` macros), and the 3rd part (%BAR) uses red, thus the 3-D effect.

```plaintext
*** CODE FOR THE TABLE ***;

** Create Box on Page **;

%Poly(90.5, 49, vipk, solid, 1);
%Polycont(160.5, 51);
%Polycont(156.5, 49);
%Polycont(95.5, 49);

%Poly(156.5, 19, vipk, solid, 1);
%Polycont(156.5, 49);
%Polycont(160.5, 51);
%Polycont(156.5, 19);
%Bar(90.5, 19, 156.5, 49, red, 0, solid);
%Bar(94, 21, 153, 47, white, 0, solid);

%Line(90.5, 49, 94.5, 51, black, 1, 1);
%Line(156.5, 19, 160.5, 21, black, 1, 1);
%Line(156.5, 49, 160.5, 51, black, 1, 1);
%Line(94.5, 51, 160.5, 51, black, 1, 1);
%Line(160.5, 21, 160.5, 51, black, 1, 1);

%Rect(94, 40.5, 153, 47, black, 1, 2);
%Rect(94, 21, 153, 40.5, black, 1, 2);

***** Titles for Table *****;
%Label(123.5, 45.75, "Demographics and", black, ., ., 1.5, hwpsl027, 5);
%Label(123.5, 44, "Characteristics", black, ., ., 1.5, hwpsl027, 5);
%Label(123.5, 42.25, "of Deceased Patients", black, ., ., 1.5, hwpsl027, 5);
%Label(101.5, 44.5, "Male", black, ., ., 1.5, hwpsl027, 5);
%Label(145.5, 44.5, "Female", black, ., ., 1.5, hwpsl027, 5);

If sex = 1 then do;
  %Label(101.5, 42.5, totsex, black, ., ., 1.25, hwpsl009, 5);
End;

If sex = 2 then do;
  %Label(145.5, 42.5, totsex, black, ., ., 1.25, hwpsl009, 5);
End;
```

*All code segment to left created in Table Editor.
Below, when placing the n's and percents in the table, the formats are set up when placing on the graph. This can be done either like this, or when the data is initially set up before going into the annotate process. We have demonstrated both processes; the N for the pie and the n's for sex on the table were both set up before starting annotate.

---- Data for Males ----

If sex = 1 then do;
  %Label(101, 38.5, compress(put(over75_p, 8.)) || "\%(" || compress(put(OVER75_N, 8.)) || ")", black, ., ., 1.25, hwps1009, 5);
  %Label(101, 35.5, compress(put(race1_p, 8.)) || "\%(" || compress(put(race1_n, 8.)) || ")", black, ., ., 1.25, hwps1009, 5);
  %Label(101, 34, compress(put(race2_p, 8.)) || "\%(" || compress(put(race2_n, 8.)) || ")", black, ., ., 1.25, hwps1009, 5);
  %Label(101, 32.5, compress(put(race3_p, 8.)) || "\%(" || compress(put(race3_n, 8.)) || ")", black, ., ., 1.25, hwps1009, 5);
  %Label(101, 29.5, compress(put(smoke_p, 8.)) || "\%(" || compress(put(smoke_n, 8.)) || ")", black, ., ., 1.25, hwps1009, 5);
  %Label(101, 26.5, compress(put(drink_p, 8.)) || "\%(" || compress(put(drink_n, 8.)) || ")", black, ., ., 1.25, hwps1009, 5);
End;

---- Data for Females ----

Continue Code. . . (same steps as Male data)

Run;

Proc ganno annotate=output name="output" gout=sug.output; Quit;

filename output "z:\tmaxwell\pharmasug\in\output.ps",

Goptions gaccess=SASGAEDT gsfmode=replace gsfname=output gsflen=132 noborder noprompt display;

Proc greplay igout=sug.output nofs tc=sashelp.templt template=whole; treplay 1: output; Delete output; Quit;

Proc greplay igout=work.gseg nofs; delete _all_; Quit;

We’ve now completed our graph, you’ll see the results on the next page. With a glance, the interpretation is clear and unmistakable.

Although we have introduced some macros available within SAS/GRAPH, there are excellent sources noted below that will describe more about annotate.

http://www.sas.com/service/techsup/sample/sample_graph.html

Kenny, S.J. (SUGI 23) How Not to Hate Annotate

Riddle, M. (SUGI 22) Two Methods to Produce Mean Plots of Clinical Trials Data: Why Simplifying with Annotate is Not an Oxymoron

Pakalniskis A., Bruce D., and Grant G. (SUGI 24) Using SAS/GRAPH Software to Compare Physician Practice

Zdeb, M. and Dairman, M. (SUGI 22) Calculating and Illustrating the Probability of Developing Cancer Using SAS and SAS/GRAPH Software
**SUMMARY**

It’s quite simple building a graph using annotate that both grabs attention and is completely self-explanatory. We typically do graphics of this nature in black and shades of gray; we added color for this presentation simply to add ‘pizzazz’ and demonstrate the flexibility. An excellent source for help when doing this is by accessing HELP in an interactive SAS session, searching for the keyword ANNOTATE.

We’ve demonstrated a very basic graphic. You can also do such things as placement of a company’s logo; additionally, multiple reports can be set up at once. An example of this would be a set of analysis tables containing 12 reports; the initial data can be set up to output by patient, site, state, territory, etc. You would then pass the data through to the annotate program via macros. This method makes the tables, through the use of graphics, much easier to read and visually appealing.

Depending on the operating system used, version of SAS, and device being used, there are different fonts, colors, special characters (IE copyright symbol), etc. available for output. The device used is in the beginning of the annotation, under ‘goptions device=’. In this example we are using a Lexmark™ color inkjet printer. To obtain more information on the devices available in SAS, use ‘Proc Gdevicex’. You can find all hardware fonts available for the device being used with this procedure. In addition to printers, SAS has a rich set of graphic ‘drivers’ that tell it exactly how the information should be prepared. SAS output can also be sent to the screen, a CATALOG entry, a camera or even a plotter.

**REFERENCES**


**CONTACT INFORMATION**

Marcia Branham
STATPROBE, Inc.
220 Lexington Green Circle, Suite 200
Lexington KY 40503
Phone: (859) 971-8899 x506
Fax: (859) 273-7929
Email: marcia.branham@lx.statprobe.com

Timothy Maxwell
STATPROBE, Inc.
220 Lexington Green Circle, Suite 200
Lexington KY 40503
Phone: (859) 971-8899 x529
Fax: (859) 273-7929
Email: timothy.maxwell@lx.statprobe.com

**PLATFORM**

The code for this paper was written using SAS Version 6.12 for Windows NT.