Using the DATASYS Option of PROC GANNO

Jade Walker
SAS Institute Inc.

The Annotate facility

The options specific to a SAS/GRAPH procedure do not always document the graph as thoroughly as you may want. Annotation is a means of customizing a graph to meet your needs. For example, in this graphic produced by PROC GCHART with the VBAR statement, the data values are represented only by the vertical bars (Fig 1). For those in your audience who prefer to see numeric representation, you can use the LABEL annotate function to add the numbers onto each bar. (Fig 2).

Annotation can be either simple or complex and can appear either on its own as with PROC GANNO or PROC GSLIDE, or with a graphics procedure such as PROC GCHART or PROC GPLOT. It can place text, lines, or shapes at your discretion through the use of annotate functions. Annotating a graph adds emphasis where you want it according to your standards of design. For example, the use of PROC GMAP can be used to display the United States (Fig 3). However, if you require a certain emphasis on one particular part of the United States (Fig 4), use of the LABEL, MOVE, and DRAW annotate functions can show this emphasis adequately. This graphic is a fairly simple example of the use of annotation. On the other hand, you can use annotation to create artistic graphics (Fig 5). This graphic shows a more complicated example using PROC GANNO. This design comes from a cross stitch pattern and consists of about 4000 instances of the BAR annotate function with one LABEL annotate function used for a title.

Annotation is performed in two phases. In the first phase, commands to annotate a graph are placed in a SAS data set, which is also called an ANNOTATE= data set. The commands require that variable names in the ANNOTATE= data set are specific to the function of annotation, such as FUNCTION, X, Y, and COLOR. In the second phase, the ANNOTATE= data set is combined with a graphics procedure through the ANNOTATE= option. The ANNOTATE= option can be global (affecting all statements in the graphics procedure) or local (affecting only that statement in which the ANNOTATE= option occurs). A graphics routine translates the annotate functions into graphics commands and draws the graph along with the annotation.

Annotation is performed based on which system you choose. An annotate system provides a method of locating annotate functions based on a certain area of the graph. The three major frameworks on which to build a graph are the axis area, the page window, and the plot window. The axis area is the area lying within the axes if the FRAME= option were used (Fig 6). Annotation performed in this area is usually dependent on the data used in the graphics procedure. The page window refers to the entire graphics area of the device (Fig 7). The BORDER option or PROC GSLIDE with the BORDER option shows the range of the page window. The plot window refers to the entire graphics area of the device except for the space that contains FOOTNOTES (Fig 8). Note that the plot window is a subset of the page window, and the axis area is a subset of the plot window (Fig 9).

In using annotation, the XSYS, YSYS, and HSYS annotate parameters describe the system of annotation. The method of the system can be absolute or relative. The viewports which can be used by annotation (page window), the window (plot window), or the data area (axis area). The window scales which can be used are in percentage or in character cells.

Absolute systems 1 (data percentage) and 2 (data value) and relative systems 7 (data relative percentage) and 8 (data relative value) reference the data area. Absolute systems 3 (window percentage) and 6 (window value) and relative systems B (window relative percentage) and C (window relative value) refer to the window. Absolute systems 3 (screen percentage) and 4 (screen value) and relative systems 9 (screen relative percentage) and A (screen relative percentage) refer to the screen.

The DATASYS option of PROC GANNO uses an implicit annotate system, one dependent upon the data given. The DATASYS option of PROC GANNO performs some analysis upon the ANNOTATE= data set. It determines the maxima and minima of X and Y in the data set and scales those values appropriately in 90% of the available graphics area. In producing these graphics, I used the annotate system of 2 for the XSYS and YSYS and 4 for the HSYS (Fig 10). The 90% scale maintains an eye-pleasing boundary around the graph. This slide shows the plot window within which PROC GANNO draws when the DATASYS option is used. PROC GANNO does separate scaling for x and y (Fig 11). This means that the x values can run 35 to 50 while the y runs from 0.0 to 0.6. There may be no (0,0) point available on the graphics area when PROC GANNO is invoked with the DATASYS option. It helps to think of the DATASYS plot window as output from GPLOT that has no axes. This slide shows the placement of the data values within the plot window, with x and y having their scale and increments.
Font Design Tools

Two applications that make use of this option are font design and chart design. While PROC GANNO is not particularly useful in the actual formation of letters, it can provide some useful tools for determining whether or not a letter is well-shaped (Fig 12). A SAS DATA step can transform a SAS data set from proper input to PROC GFONT to a SAS data set of annotate functions. Another DATA step can transform annotate functions that form letters into a SAS data set that can be used as input to PROC GFONT.

Once each letter is in an ANNOTATE= data set, the font tools can be exercised. In many cases, constructing a font requires several days of typing in data, then running PROC GFONT (Fig 13). The graph produced by PROC GFONT, however, does not produce an adequate rendition. On a low- to medium-resolution device, it does not sufficiently emphasize the outlier points. In addition, the appearance of so many letters on one screen has a jarring effect and does not allow for effective examination of any single letter. By creating an ANNOTATE= data set from the input set to PROC GFONT, one can examine the letters singly and at much greater magnification than that of PROC GFONT (Fig 14). It is easy to see if the lines meet, are of uniform width and follow the desired shape of the letter.

The code producing this graph is actually a generalized macro that selects by character and produces a graph of that particular character.

Other macros can be used to create common references lines so that the characters produced can be compared. One method is to use width reference lines to see if two characters that should be the same width are actually the same (Fig 15). In this graphic, the width lines are hardcoded to my width speciﬁcations. The starting and ending width could just as well be parameters to the macro. As you can see, the capital A is a bit wider than planned (Fig 16). In this graphic, the capital E is a little too narrow. Through iterative reﬁnement, these two letters can be given the same width.

In the same manner, you can fashion height reference lines (Fig 17). The height reference lines in this example are also hardcoded, with different heights required for the baseline, the descenders, the miniscule (small) letters, the majuscule (capital) letters, and the special letters like the miniscule t, which in this font is larger than the miniscule but smaller than the majuscule letters. This allows the letters to be compared for uniform height.

If, in designing the font, one needs access to the data values used in constructing the letter, the PTANNO macro displays the location of the data values along with the letter (Fig 18). This macro allows the font designer to determine by how much to change the data points. Occasionally, the points will be too close together to be able to decipher the (x, y) coordinates. In this case, the font designer should use a subsetting IF statement in the DATA step which constructs the data points to build only those data points he needs.

Another useful method to ensure uniformity among letters is to compare them side by side (Fig 19). This slide shows the capital letters A and E together. This macro takes as its arguments the name of the first character, the name of the second character, and the width of the first character plus some space between the characters.

If modifications have been made to the ANNOTATE= data set, a DATA step is needed to return the information back into a form acceptable to PROC GFONT (Fig 20). The LP variable is written based on the FUNCTION variable in the ANNOTATE= data set.

Chart Design

Using a different view of the available graphics area, you can create charts and/or plots quite easily. You can make charts using the DATASYS option of PROC GANNO if the data values are numeric, discrete, limited, and ordered. Continuous functions are displayed using this method because labels for continuous functions would be hard to place. Character values can be used if they can be converted easily into numeric values (for example, JAN FEB MAR can become 1, 2, 3 quite easily). In addition, some cases of numeric values need to have equal intervals between to create, for example, equally spaced bars on a graph.

This graphic shows an unannotated HBAR chart (Fig 21). The data values available to PROC GANNO are contained within the axis area and this part of the screen is the only section accessible to the data value system (system 2 in this instance). This graphic shows the same chart annotated with horizontal reference lines at some of the available data values (Fig 22). If the value 0 were used as a Y value, for example, the ANNOTATE facility would mark that observation as invalid because that data value is not available on this chart. This graphic also adds the vertical reference lines, forming a grid over the graph (Fig 23).

To construct a HBAR chart using the DATASYS option of PROC GANNO, the response axis becomes the scale in the x direction, and the midpoint axis becomes the scale for the y direction. If a data value summary, average, or maximum is needed, use the output data set
generated from either PROC MEANS or PROC MEANS command and connected with %MOVE and %DRAW commands (Fig 29). Next, the maximum, minimum, and range of x and y were computed using PROC MEANS. In the ANNOTATE= data set for the axes, the x axis was drawn using the coordinates \( x_{\text{min}} \) and \( x_{\text{max}} \), and the y axis was drawn using the coordinates \( y_{\text{min}} \) and \( y_{\text{max}} \). An interval was chosen to place the tick marks along the axis using the %LABEL command with a hardware plus sign in a DO loop for each axis. In this example, the axes were arbitrarily divided into tenths. The values next to the tick marks were also generated in the same DO loop which drew the tick marks (Fig 30). After completing both the data points and axis to your specification, combine the data sets with the SET command in the DATA step and produce the graph from the resulting ANNOTATE= data set (Fig 31).

This graphic was shown last year at SUGI 10 for a SAS/ETS® paper. (Fig 32.) It was done using PROC GSLIDE with the plot window annotate system and took a lot of effort on the part of the developer to create. It represents the ideal or textbook case of collected data upon which you can make a forecast. A replica of this slide can be created using the DATASYS option of PROC CANNO (Fig 33).

This graphic was produced using the years for its Y axis and, with a little adjustment, the values X1-X4 and Y1-Y4 for its Y axis. %BAR commands create the actual and assumed data bars as well as the frame. The labels were added later after the body of the graph was complete. One can usually guess the required location of the labels once the graph has been displayed. If the labels are required only once on a graph, use the IF \_N_ = 1 logic to place them. If the labels are directly related to the x or y values, put the %LABEL command inside the %ANNO command after the processing of these values. The year labels are done in this fashion. This adaptation of the original slide to a casual hour to write the code. In addition, the original slide took 3.46 seconds and 1890K to create. The slide produced using the DATASYS option took 3.39 seconds and 1890K. So taking into account the code development time, the graph creation time, and the memory used, the method using the DATASYS option is quite practical.

Another application of this technique was used to create a cartesian plot. The current version of PROC GLOT does not produce plots with four quadrants, so this method was developed. First, the points are created using the %LABEL command and connected with %MOVE and %DRAW commands (Fig 29). In the ANNOTATE= data set for the axes, the x axis was drawn using the coordinates \( x_{\text{min}} \) and \( x_{\text{max}} \), and the y axis was drawn using the coordinates \( y_{\text{min}} \) and \( y_{\text{max}} \), and the y axis was drawn using the coordinates \( y_{\text{min}} \) and \( y_{\text{max}} \). An interval was chosen to place the tick marks along the axis using the %LABEL command with a hardware plus sign in a DO loop for each axis. In this example, the axes were arbitrarily divided into tenths. The values next to the tick marks were also generated in the same DO loop which drew the tick marks (Fig 30). After completing both the data points and axis to your specification, combine the data sets with the SET command in the DATA step and produce the graph from the resulting ANNOTATE= data set (Fig 31).

This graphic was shown last year at SUGI 10 for a SAS/ETS® paper. (Fig 32.) It was done using PROC GSLIDE with the plot window annotate system and took a lot of effort on the part of the developer to create. It represents the ideal or textbook case of collected data upon which you can make a forecast. A replica of this slide can be created using the DATASYS option of PROC CANNO (Fig 33).

This graphic was produced using the years for its Y axis and, with a little adjustment, the values X1-X4 and Y1-Y4 for its Y axis. %BAR commands create the actual and assumed data bars as well as the frame. The labels were added later after the body of the graph was complete. One can usually guess the required location of the labels once the graph has been displayed. If the labels are required only once on a graph, use the IF \_N_ = 1 logic to place them. If the labels are directly related to the x or y values, put the %LABEL command inside the %ANNO command after the processing of these values. The year labels are done in this fashion. This adaptation of the original slide to a casual hour to write the code. In addition, the original slide took 3.46 seconds and 1890K to create. The slide produced using the DATASYS option took 3.39 seconds and 1890K. So taking into account the code development time, the graph creation time, and the memory used, the method using the DATASYS option is quite practical.

Another application of this technique was used to create a cartesian plot. The current version of PROC GLOT does not produce plots with four quadrants, so this method was developed. First, the points are created using the %LABEL command and connected with %MOVE and %DRAW commands (Fig 29). Next, the maximum, minimum, and range of x and y were computed using PROC MEANS. In the ANNOTATE= data set for the axes, the x axis was drawn using the coordinates \( x_{\text{min}} \) and \( x_{\text{max}} \), and the y axis was drawn using the coordinates \( y_{\text{min}} \) and \( y_{\text{max}} \), and the y axis was drawn using the coordinates \( y_{\text{min}} \) and \( y_{\text{max}} \). An interval was chosen to place the tick marks along the axis using the %LABEL command with a hardware plus sign in a DO loop for each axis. In this example, the axes were arbitrarily divided into tenths. The values next to the tick marks were also generated in the same DO loop which drew the tick marks (Fig 30). After completing both the data points and axis to your specification, combine the data sets with the SET command in the DATA step and produce the graph from the resulting ANNOTATE= data set (Fig 31).
In this lifespan graph, the numeric value used along the y axis is the observation number. However, in order to retain the alphabetical order of the observations, the observation number was subtracted from a constant which inverted the order along the y axis. The values for the x axis are years which are already numeric. A %RECT command draws the frame and a combination of %MOVES and %DRAWs make the horizontal reference lines (Fig 34).

Instead of showing a slide made from a listing using the hardware characters of the device, you could present the figures in this manner. The years are the x values and specially coded numbers for the department are the y values. This slide gets the numbers across but also uses the color and spacing of presentation graphics (Fig 35).

Font design and chart design are but two applications available through the DATASYS option. The font design application allows you to form characters of uniform shape, height, and width. The chart design application allows a more customized type of chart without relying on the usual SAS/GRAPH formats. This option gives users more flexibility and versatility in your presentation graphics.
Announcing SAS/GRAPH Version

Fig 7

This is a page window

Fig 5

This is an axis area

Fig 6

This is a plot window

Fig 4

Fig 8

Titles and footnotes

make the plot window smaller
Use systems 1, 2, 7, and 8

Use systems 5, 6, B, and C

Use systems 3, 4, 9, and A

Fig 11

Fig 10

PROC GFONT input

LP = P?

No

Yes

MOVES and DRAWs

POLYS and POLYCONTs

Plot Window
### Structure of Assumption Data Set

*(Textbook Case)*

<table>
<thead>
<tr>
<th>Year</th>
<th>Y1</th>
<th>Y2</th>
<th>Y3</th>
<th>Y4</th>
<th>X1</th>
<th>X2</th>
<th>X3</th>
<th>X4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1977</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1978</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1979</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1980</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1981</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1982</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1983</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1984</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1985</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1986</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1987</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1988</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1989</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1990</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Legend:**
- **Endogenous**
- **Exogenous**
- **Actual**
- **Assumption**

*Fig. 31*

*Fig. 30*
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>TOOLS</td>
<td>$2,000</td>
<td>$1,250</td>
<td>$1,000</td>
<td>$1,100</td>
</tr>
<tr>
<td>PANTS</td>
<td>$900</td>
<td>$1,000</td>
<td>$1,100</td>
<td>$1,000</td>
</tr>
<tr>
<td>REPAIR</td>
<td>$1,200</td>
<td>$1,100</td>
<td>$1,250</td>
<td>$1,800</td>
</tr>
</tbody>
</table>

Fig 35