

To Annotate or Not to Annotate, There Should Be No Question!

Keith Cranford, Marquee Associates, LLC

Abstract

The Annotate Facility within SAS/GRAPH is a powerful tool for enhancing your graphs. This is one of the tools that distinguishes SAS/GRAPH from other graphing software. This tutorial outlines the basic concepts of the Annotate Facility. Annotate macros, which are provided with SAS/GRAPH software and facilitate the use of annotation, are also discussed.

Introduction

The Annotate Facility within SAS/Graph provides a tool for enhancing graphics in data-driven form. Some of these enhancements include adding values to points on a line graph or histogram, providing custom labelling, or producing custom graphics from scratch. The Annotate Facility may be used in conjunction with other SAS/Graph procedures, or it can be used with PROC GANNO or PROC GSLIDE to control all the graphics.

Annotate Data Set

The Annotate Facility is comprised of a data set which contains graphic instructions. This data set consists of specially named variables, which define these instructions. By assigning values to these variables for each observation, you control

- Coordinate system that is used
- Task that is to be performed
- Position of the graphic objects
- Definition and position of graphic text

The coordinate system is controlled by three variables, XSYS, YSYS and HSYS. Each of these variables is a 1-byte character field. The XSYS variable indicates the horizontal axis coordinate system, the YSYS variable the vertical axis, and the HSYS variable the coordinate system used by the SIZE variable. There are three different coordinate systems:

- **Data:** uses the same coordinate system as the graphics
- **Procedure:** includes all the graphics output area between the titles and footnotes
- **Display:** includes the entire graphics output area

Table 1 gives some of the possible values of the coordinate system variables in terms of the different coordinate systems. Generally, it is easiest to work with the percentage values, since cell values can vary based on device.

Table 1: Coordinate System Values

XSYS/YSYS/HSYS value	Coordinate System	Range
'1'	Data %	0% to 100% of Axis
'2'	Data value	Axis Min to Axis Max
'3'	Display %	0% to 100% of Display
'4'	Display value (in cells)	0 to Edge of Display
'5'	Procedure %	0% to 100% of Procedure
'6'	Procedure value (in cells)	0 to Edge of Procedure

The position of the graphics or text within the specified coordinate system is controlled by the X and Y variables. These variables control the horizontal and vertical axis locations, respectively. They are numeric variables with the range of values indicated in the table above.

The FUNCTION variable indicates the action to be taken, i.e., what you want to do. This is an 8-byte character variable with a default value of LABEL. Some of the possible values are

- LABEL: draws text at the specified location
- MOVE: moves the positional pointer to a specified location
- DRAW: draws a line from the current location to a specified location
- BAR: draws and, optionally, fills a rectangle
- POINT: draws a point
- POLY: begins drawing a polygon
- POLYCONT: continues drawing polygon
- SYMBOL: draws a symbol

The values of additional variables control other attributes of the Annotate instruction. These variables have fixed names, types and lengths, control different aspects of the graphics element depending on the value of FUNCTION, and are assigned default values if not included in the data set. These variables, along with their characteristics and default values, are given in Table 2.

Table 2: Descriptions of Annotate Variables

Variable	Type (length)	Default
ANGLE	Num	0
COLOR	Char (8)	First in device list
LINE	Num	1
POSITION	Char (1)	'5'
ROTATE	Num	0
SIZE	Num	1
STYLE	Char (8)	Depends on FUNCTION
TEXT	Char (<= 200)	Blank
WHEN	Char (1)	'B' (before)

Use of these attribute variables and the other Annotate variables can best be illustrated with the LABEL function. To place text at a specific location, assign FUNCTION the value 'LABEL' and assign values to the attribute variables to control the look of the text. Table 3 gives the possible and default values for these variables.

Table 3: Possible Values of Annotate Variables

Variable	Possible Values	Default Value
X,Y	Depends on coordinate system	Last values of X,Y
XSYS,YSYS,ZSYS	'1' to 'C'	'4'
POSITION	'1' to 'F'	'5'
SIZE	> 0	1
STYLE	Valid font name	'NONE'
COLOR	Valid color name	First in device list
ANGLE	-90 to 90	0
ROTATE	0 to 360	0
TEXT	Any text	Blank
WHEN	'A' or 'B'	'B'

The POSITION variable is coded to align the text. Table 4 provides possible values and alignments for this variable.

Table 4: POSITION Values

Vertical Pos	Right Align	Center	Left Align
One cell above	'1'	'2'	'3'
Half cell above	'A'	'B'	'C'
Centered	'4'	'5'	'6'
Half cell below	'D'	'E'	'F'
One cell below	'7'	'8'	'9'

The ANGLE variable specifies an angle of rotation of the entire text from horizontal. For example, ANGLE=-90 would print the text vertically (reading the text by tilting your head to the right). The ROTATE variable specifies an angle of rotation for each character in the text starting from the orientation after the text has been angled. Thus, ROTATE=90 would rotate each letter in the ANGLE=-90 example back to horizontal, so the text can be read down the page.

The WHEN variable specified when the text is printed, either before ('B') or after ('A') the graph is displayed. This determines what is displayed when there is an overlap between the text and graphics. The STYLE variable for the LABEL function specifies the font to be used. These can be SAS-supplied fonts, such as Swiss, Duplex and Simplex, or True type fonts, which are specified in quotes.

Annotate Macros

Annotate variables can be populated either through assignment statements, or by using annotate macros that are available through SAS/GRAPH. Each macro is controlled by a set of parameters, which determine the values assigned to the Annotate variables. The macros are accessible upon issuing the macro %annomac, which activates these macros. Some of the common macros listed below are more fully documented

in the SAS/GRAPH Software: Reference, Volume 1, or in the online documentation.

- **%LABEL:** draws text
- **%MOVE:** moves without drawing
- **%DRAW:** draws a line from the previous point
- **%LINE:** draws a line between to points
- **%BAR:** draws a bar
- **%CIRCLE:** draws a circle
- **%SYSTEM:** sets the coordinate system
- **%SEQUENCE:** specifies when to draw

Each of these macros are controlled by a set of parameters. These parameters determine the values of the Annotate variables needed to produce the desired effect. Since a single macro call can produce multiple observations in the Annotate data set, this also makes for succinct code.

Use with SAS/Graph Procedures

Annotate data sets can be used with SAS/Graph procedures to enhance the standard output of these procedures. Some simple uses are to add data values to line plots or bar charts and to add custom text to any graph. A more complex example is to use Annotate to overlay a line plot on a bar chart, or add a data table to a graph. In these cases, the data set that is used as input to the SAS/GRAPH procedure is also used to create the Annotate data set.

For example, you may want to label the points on a line plot. To do this, you need to indicate what function you want to perform (FUNCTION = 'LABEL'), where you want to place the label (X = horizontal-axis variable value, Y = vertical-axis variable value, and POSITION = centered above the point), and what you want to label (TEXT = vertical-axis variable value). The following code would accomplish this.

```
data data ;
    input X Y @@ ;
    datalines ;
5 8 6 10 7 11 8 14
;
run ;

data anno ;
    set data ;
    length function $ 8 text $ 15 ;
    retain xsys ysys '2' when 'a' ;
    function = 'LABEL' ;
    x = x ;
    y = y ;
    text = left(put(y, best.)) ;
    style = 'swissb' ;
    size = 2 ;
    position = '2' ;
run ;

goptions htext=2 ftext=swissb ;
axis1 order=6 to 15 by 3 minor=none ;
```

```

axis2 order=5 to 8 minor=none
      offset=(5 pct,) ;
proc gplot data=data anno=anno ;
  plot y*x / vaxis=axis1
            haxis=axis2 ;
  title 'Figure 1' ;
  symbol i=join value=star ;
run ;
quit ;

```

This code would produce the graph in Figure 1.

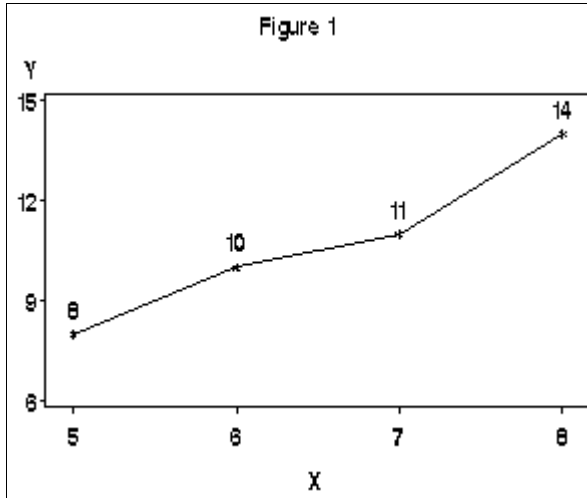


Figure 1: PROC Gplot with Annotate

In the previous example, the assignment statements could be replaced by Annotate macros.

```

%annomac ;
data anno ;
  set data ;
  length function $ 8 text $ 15 ;
  retain xsys ysys '2' when 'a' ;
  %label(x,y,left(put(y,best.)),black,
        0,0,2,swissb,2) ;
run ;

```

Also, the retain statement could be replaced by

```

%system(2,2,4) ;
%sequence(A) ;

```

Instead of labelling the data within the graph, you might include a table below the graph. In this case, you would use the display coordinate system and fix the position of the axes. This will allow you to know where to place the table in relation to the graph and the horizontal axis values. The following code would produce the graph in Figure 2.

```

%annomac ;
data anno ;
  set data(rename=(y=y1)) ;
  length function $ 8 text $ 15 ;
  retain xsys ysys '3' when 'a' ;

```

```

if _n_=1 then do ;
  %label(5,4,'X',black,0,0,2,swissb,B) ;
  %line(10,10,90,10,black,1,.5) ;
  %line(90,10,90,3,black,1,.5) ;
  %line(90,3,10,3,black,1,.5) ;
  %line(10,3,10,10,black,1,.5) ;
  do i=1 to 4 ;
    %line(10+i*20,3,10+i*20,10,
          black,1,.5) ;
  end ;
end ;
j+1 ;
%label(j*20,4,left(put(y1,best.)),
      black,0,0,2,swissb,B) ;
run ;

goptions htext=2 ftext=swissb;
axis1 order=6 to 15 by 3 minor=none ;
axis2 order=5 to 8 by 1 minor=none
      offset=(10pct,) origin=(10pct,20pct)
      length=80 label=none;
proc gplot data=data anno=anno ;
  plot y * x / vaxis=axis1 haxis=axis2 ;
  title 'Figure 2' ;
  symbol i=join value=star;
run ;
quit ;

```

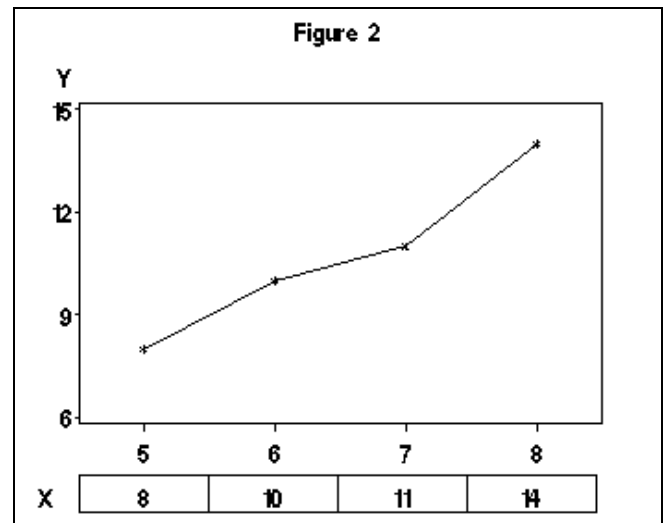


Figure 2: PROC Gplot with Table

Use with GANNO or GSLIDE Prodecures

The Annotate Facility can also be used to produce custom graphics using the GANNO or GSLIDE procedures. In this case the Annotate data set will be used to issue all the instructions to produce a graphic. This may be in the form of a more traditional graphic output such as a line plot, bar chart or pie chart, or it may be a custom graphics table.

For example, the Annotate Facility could be used to produce an invoice containing a customers address information as well as purchase detail. Suppose customer data was stored in a data set called CUSTOMER and the invoice detail in a data set called INVOICE. These two data sets are then linked by the customer_id variable. The following code would produce the invoice graphic output in Figure 2.

```
proc sql ;
  create table detail as
  select *
  from customer as a, invoice as b
  where a.customer_id=b.customer_id and
        invoice_no=1 ;
quit ;

%annomac ;
data anno ;
  set detail end=last ;
  length text $ 30 ;
  retain xsys ysys '2' when 'a' ;
  if _n_=1 then do ;
    %label(5,95,name,black,0,0,2,'Arial',C) ;
    %label(94,95,'Inv #'||put(invoice_no,3.),
           black,0,0,2,'Arial',A) ;

    %label(5,90,address,black,0,0,2,
           'Arial',C) ;
    %label(5,85,city_st_zip,black,
           0,0,2,'Arial',C) ;
    %line(5,80,95,80,black,1,.5) ;
    %line(5,72,95,72,black,1,.5) ;
    %line(5,28,95,28,black,1,.5) ;
    %line(95,80,95,20,black,1,.5) ;
    %line(95,20,5,20,black,1,.5) ;
    %line(5,20,5,80,black,1,.5) ;
    %label(6,74,'Item',black,0,0,2,
           'Arial',C) ;
    %label(30,74,'Quantity',black,0,0,2,
           'Arial',C) ;
    %label(70,74,'Unit $',black,0,0,2,
           'Arial',A) ;
    %label(94,74,'Total $',black,0,0,2,
           'Arial',A) ;
    %label(6,30,'Tax',black,0,0,2,'Arial',C) ;
    %label(6,21,'Total',black,0,0,2,
           'Arial',C) ;
    y1=70 ;
  end ;
  y1+(-5) ;

  %label(6,y1,item,black,0,0,2,'Arial',C) ;
  %label(40,y1,put(quantity,8.0),black,0,0,2,
         'Arial',A) ;
  %label(70,y1,put(unitcost,8.2),black,0,0,2,
         'Arial',A) ;
  %label(94,y1,put(quantity*unitcost,8.2),
```

```
        black,0,0,2,'Arial',A) ;
  totcost+(quantity*unitcost) ;
  if last then do ;
    tax=round(totcost*.08,.01) ;
    %label(94,30,put(tax,8.2),black,0,0,2,
           'Arial',A) ;
    totcost + tax ;
    %label(94,21,put(totcost,8.2),black,0,0,2,
           'Arial',A) ;
    %label(5,12,'Balance Due Upon Receipt',
           black,0,0,2,'Arial',C) ;
  end ;
run ;

proc ganno anno=anno ;
run ;
quit ;
```

Keith 123 Easy Street River City, IA 50001			Inv # 1
Item	Quantity	Unit \$	Total \$
T-shirt	2	12.00	24.00
Socks	1	5.00	5.00
Slacks	3	30.00	90.00
Tax			9.52
Total			128.52
Balance Due Upon Receipt			

Figure 3: PROC GANNO Example

Some Helpful Suggestions

The following suggestions or guidelines can help make using the Annotate facility easier and more useful.

- Let SAS/GRAPH procedures do as much as possible
- Break the annotation into groups
- Use the ORIGIN= option to anchor the graph

The first suggestion is a time saver. The SAS/GRAPH procedures provide a lot of flexibility and features. Take advantage of this functionality. Before adding any annotation the basic graph should first be refined. In most cases, this is enough and no annotation is

needed; however, when enhancements are needed the amount of annotation is kept to a minimum.

This leads to the second suggestion. It is best to work in pieces by defining groups of annotation, such as a table, title or labeling, and then add one annotation at time. Doing this will allow you to concentrate development in one area and will also help in debugging.

A helpful hint is using the ORIGIN= option on the AXIS statement. This allows you to anchor your graph, which helps in placing annotation outside the graph, such as tables.

Conclusion

The Annotate Facility in SAS/Graph is a very powerful tool that can transform your ordinary SAS/Graph output into something extraordinary. Its use hinges on an understanding of the Annotate data set, which has a definite structure. Once this is accomplished, there really should be no question as to whether it should be used.

Keith Cranford
Marquee Associates, LLC
2101 Wychwood Dr.
Austin, TX 78746
kcranford@marquee-assoc.com
www.marquee-assoc.com