

Paper 154-2010

## Using PROC SGPLOT for Quick High-Quality Graphs

Susan J. Slaughter, Avocet Solutions, Davis, CA  
Lora D. Delwiche, University of California, Davis, CA

### ABSTRACT

New with SAS® 9.2, ODS Graphics introduces a whole new way of generating high-quality graphs using SAS. With just a few lines of code, you can add sophisticated graphs to the output of existing statistical procedures, or create stand-alone graphs. The SGPLOT procedure produces a variety of graphs including bar charts, scatter plots, and line graphs. This paper shows how to produce several types of graphs using PROC SGPLOT, and how to create paneled graphs by converting PROC SGPLOT to PROC SGPANEL. This paper also shows how to send your graphs to different ODS destinations, how to apply ODS styles to your graphs, and how to specify properties of graphs, such as format, name, height, and width. Last, this paper shows how to use the SAS/GRAPH® ODS Graphics Editor to make one-time changes to graphs..

### INTRODUCTION

When ODS Graphics was originally conceived, the goal was to enable statistical procedures to produce sophisticated graphs tailored to each specific statistical analysis, and to integrate those graphs with the destinations and styles of the Output Delivery System. In SAS 9.2, over 60 statistical procedures have the ability to produce graphs using ODS Graphics. A fortuitous side effect of all this graphical power has been the creation of a set of procedures for creating stand-alone graphs (graphs that are not embedded in the output of a statistical procedure). These procedures all have names that start with the letters SG (SGPLOT, SGSCATTER, SGPANEL, and SGRENDER).

This paper focuses on one of those procedures, the SGPLOT procedure. PROC SGPLOT produces many types of graphs. In fact, this one procedure produces 16 different types of graphs. PROC SGPLOT creates one or more graphs and overlays them on a single set of axes. (There are four axes in a set: left, right, top, and bottom.) Other SG procedures create panels with multiple sets of axes, or render graphs using custom ODS graph templates. Because the syntax for SGPLOT and SGPANEL are so similar, we also show an example of SGPANEL which produces a panel of graphs all using the same axis specifications.

This paper was written using SAS 9.2 Phase 2, but almost all the features discussed here also work in SAS 9.2 Phase1. For those features that are new with Phase 2, we note the differences in their descriptions.

### ODS GRAPHICS VS. TRADITIONAL SAS/GRAPH

To use ODS Graphics you must have SAS/GRAPH software which is licensed separately from Base SAS. Some people may wonder whether ODS Graphics replaces traditional SAS/Graph procedures. No doubt, for some people and some applications, it will. But ODS Graphics is not designed to do everything that traditional SAS/Graph does, and does not replace it. For example, ODS Graphics does not create contour plots; for contour plots you need to use traditional SAS/GRAPH.

Here are some of the major differences between ODS Graphics and traditional SAS/GRAPH procedures.

#### Traditional SAS/GRAPH

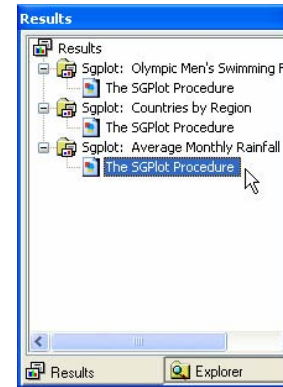
- Graphs are saved in SAS graphics catalogs
- Graphs are viewed in the Graph window
- Can use GOPTIONS statements to control appearance of graphs

#### ODS Graphics

- Produces graphs in standard image file formats such as PNG and JPEG
- Graphs are viewed in standard viewers such as a web browser for HTML output
- GOPTIONS statements have no effect

## VIEWING ODS GRAPHICS

When you produce ODS Graphics in the SAS windowing environment, for most output destinations the Results Viewer window opens to display your results. However, when you use the LISTING destination, graphs are not automatically displayed. You can always view graphs, regardless of their destination, by double-clicking their graph icons in the Results window.



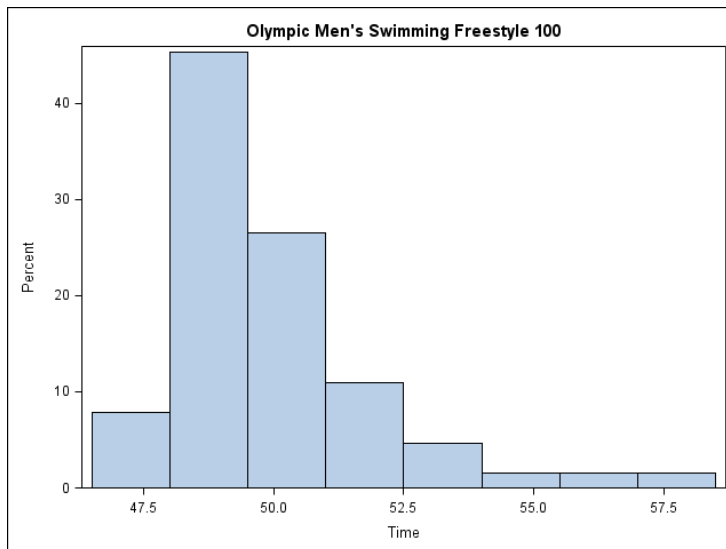
## EXAMPLES

The following examples show a small sample of the types of graphs the SGPLOT procedure can produce.

### HISTOGRAMS

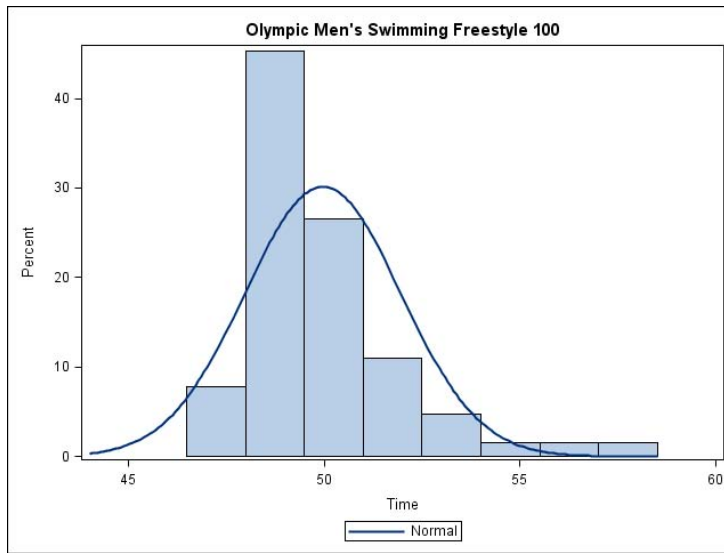
Histograms show the distribution of a continuous variable. The following PROC SGPLOT uses data from the preliminary heats of the 2008 Olympic Men's Swimming Freestyle 100 m event. The histogram shows the distribution of the variable, TIME, which is the time in seconds for each swimmer.

```
* Histograms;
PROC SGPLOT DATA = Freestyle;
  HISTOGRAM Time;
  TITLE "Olympic Men's Swimming Freestyle 100";
RUN;
```



The next PROC SGPLOT uses a DENSITY statement to overlay a density plot on top of the histogram. The default density plot is the normal distribution. When overlaying plots, the order of the statements determines which plot is drawn on top. The plot resulting from the first statement will be on the bottom, followed by the second, and so on. Care must be taken to make sure that the subsequent plots do not obscure the first.

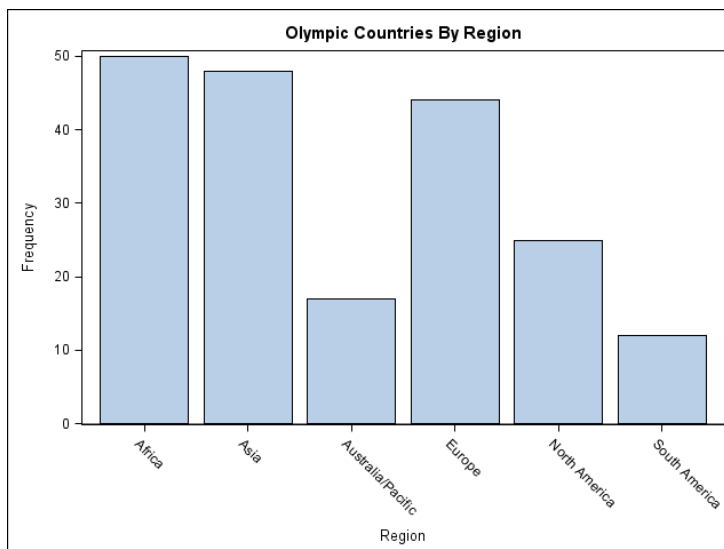
```
PROC SGPLOT DATA = Freestyle;
  HISTOGRAM Time;
  DENSITY Time;
  TITLE "Olympic Men's Swimming Freestyle 100";
RUN;
```



## BAR CHARTS

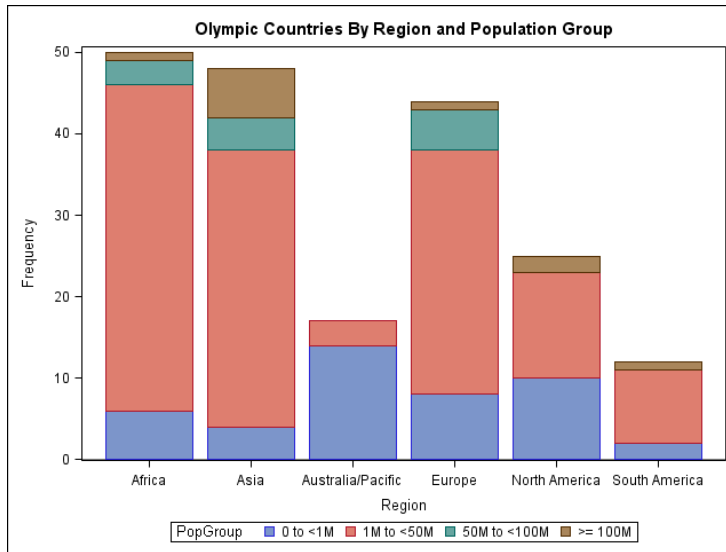
Bar charts show the distribution of a categorical variable. This code uses a VBAR statement to create a vertical bar chart of the variable REGION. The chart shows the number of countries in each region that participated in the 2008 Olympics.

```
* Bar Charts;
PROC SGPLOT DATA = Countries;
  VBAR Region;
  TITLE 'Olympic Countries by Region';
RUN;
```



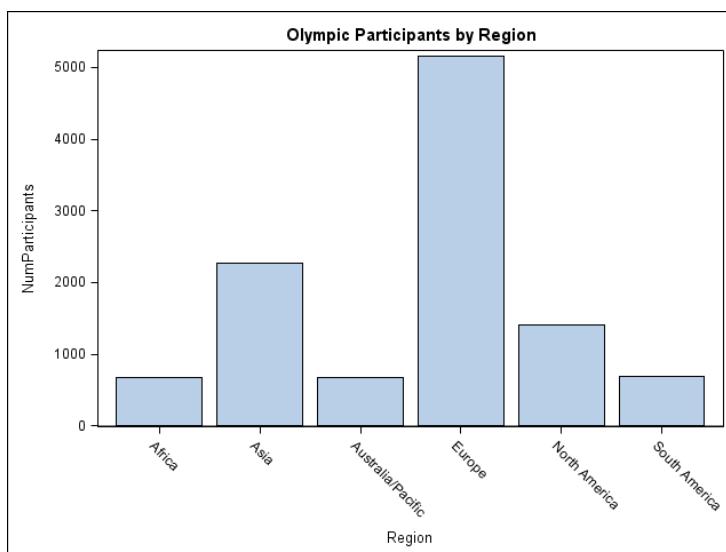
This bar chart is like the first one except that the bars have been divided into groups using the GROUP= option. The grouping variable is a categorical variable named POPGROUP. The GROUP= option can be used with many SGLOT statements (see Table 1).

```
PROC SGLOT DATA = Countries;
  VBAR Region / GROUP = PopGroup;
  TITLE 'Olympic Countries by Region and Population Group';
RUN;
```



In the following code, the GROUP= option has been replaced with a RESPONSE= option. The response variable is NUMPARTICIPANTS, the number of participants in the 2008 Olympics from each country. Now each bar represents the total number of participants for a region.

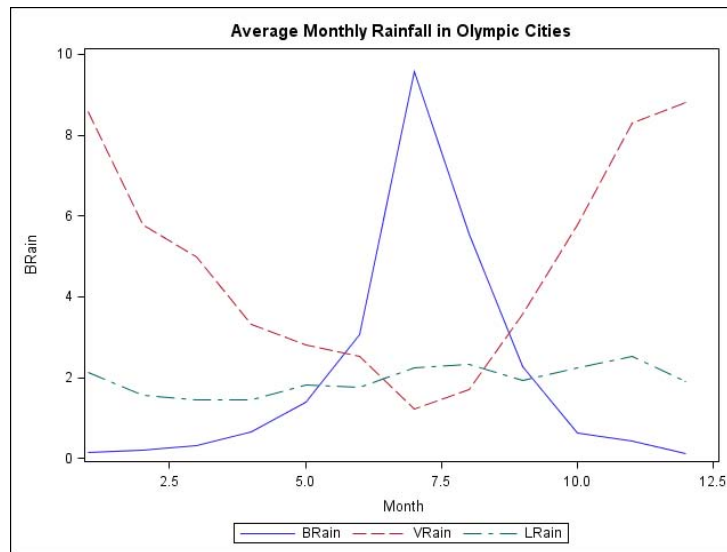
```
PROC SGLOT DATA = Countries;
  VBAR Region / RESPONSE = NumParticipants;
  TITLE 'Olympic Participants by Region';
RUN;
```



## SERIES PLOTS

In a series plot, the data points are connected by a line. This example uses the average monthly rainfall for three cities, Beijing, Vancouver, and London. Three SERIES statements overlay the three lines. Data for series plots must be sorted by the X variable. If your data are not already in the correct order, then use PROC SORT to sort the data before running the SGPLOT procedure.

```
* Series plot;
PROC SGPLOT DATA = Weather;
  SERIES X = Month Y = BRain;
  SERIES X = Month Y = VRain;
  SERIES X = Month Y = LRain;
  TITLE 'Average Monthly Rainfall in Olympic Cities';
RUN;
```



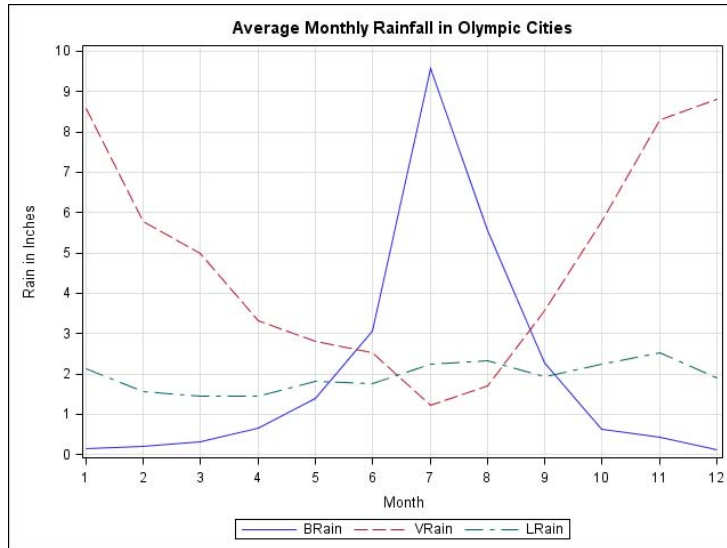
## EMBELLISHING GRAPHS

So far the examples have shown how to create basic graphs. The remaining examples show statements and options you can use to change the appearance of your graphs.

### XAXIS AND YAXIS STATEMENTS

In the preceding series plot, the variable on the X axis is Month. The values of Month are integers from 1 to 12, but the default labels on the X axis have values like 2.5. In the following code, the option TYPE = DISCRETE tells SAS to use the actual data values. Other options change the axis label and set values for the Y axis, and add grid lines.

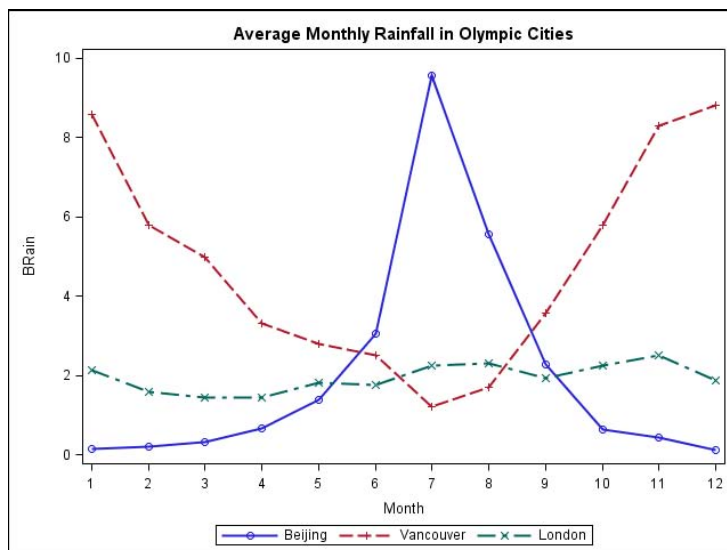
```
* Plot with XAXIS and YAXIS;
PROC SGPLOT DATA = Weather;
  SERIES X = Month Y = BRain;
  SERIES X = Month Y = VRain;
  SERIES X = Month Y = LRain;
  XAXIS TYPE = DISCRETE GRID;
  YAXIS LABEL = 'Rain in Inches' GRID VALUES = (0 TO 10 BY 1);
  TITLE 'Average Monthly Rainfall in Olympic Cities';
RUN;
```



### PLOT STATEMENT OPTIONS

Many options can be added to plot statements. For these SERIES statements, the options LEGENDLABEL=, MARKERS, AND LINEATTRS= have been added. The LEGENDLABEL= option can be used with any of the plot statements, while the MARKERS and LINEATTRS= options can only be used with certain plot statements (see Table 1).

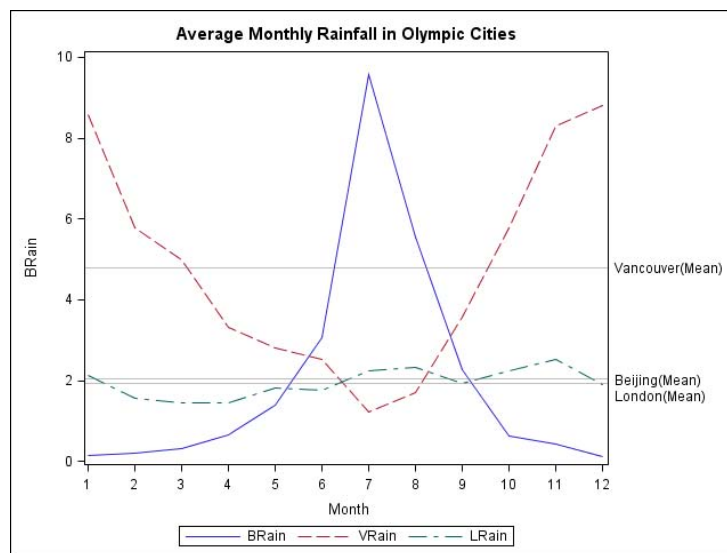
```
* Plot with options on plot statements;
PROC SGPLOT DATA = Weather;
  SERIES X = Month Y = BRain / LEGENDLABEL = 'Beijing'
    MARKERS LINEATTRS = (THICKNESS = 2);
  SERIES X = Month Y = VRain / LEGENDLABEL = 'Vancouver'
    MARKERS LINEATTRS = (THICKNESS = 2);
  SERIES X = Month Y = LRain / LEGENDLABEL = 'London'
    MARKERS LINEATTRS = (THICKNESS = 2);
  XAXIS TYPE = DISCRETE;
  TITLE 'Average Monthly Rainfall in Olympic Cities';
RUN;
```



## REFLINE STATEMENT

Reference lines can be added to any type of graph. In this case, lines have been added marking the average rainfall per month for the entire year for each city. The `TRANSPARENCY=` option on the `REFLINE` statement specifies that the reference line should be 50% transparent. The `TRANSPARENCY` option can also be used with many other plot statements (see Table 1).

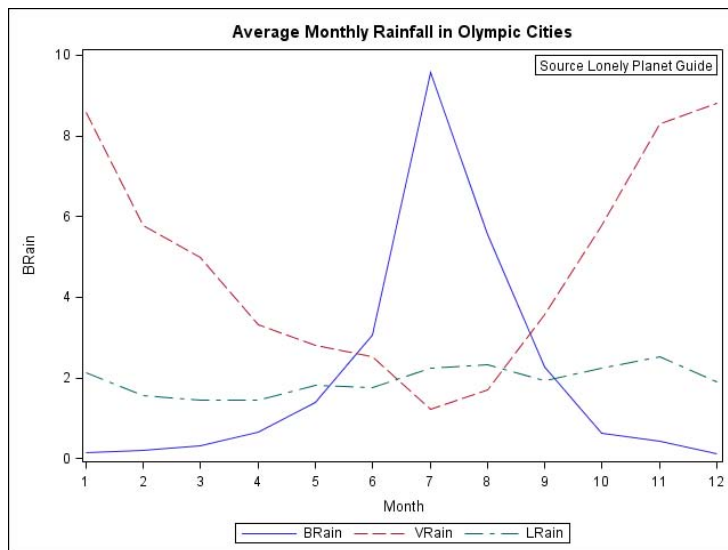
```
* Plot with REFLINE;
PROC SGPLOT DATA = Weather;
  SERIES X = Month Y = BRain;
  SERIES X = Month Y = VRain;
  SERIES X = Month Y = LRain;
  XAXIS TYPE = DISCRETE;
  REFLINE 2.03 4.78 1.94 / TRANSPARENCY = 0.5
    LABEL = ('Beijing(Mean)' 'Vancouver(Mean)' 'London(Mean)');
  TITLE 'Average Monthly Rainfall in Olympic Cities';
RUN;
```



## INSET STATEMENT

The `INSET` statement allows you to add descriptive text to graphs. Insets can be added to any type of graph.

```
* Plot with INSET;
PROC SGPLOT DATA = Weather;
  SERIES X = Month Y = BRain;
  SERIES X = Month Y = VRain;
  SERIES X = Month Y = LRain;
  XAXIS TYPE = DISCRETE;
  INSET 'Source Lonely Planet Guide' / POSITION = TOPRIGHT BORDER;
  TITLE 'Average Monthly Rainfall in Olympic Cities';
RUN;
```



## SGPLOT PROCEDURE SYNTAX

The four tables spread over the next five pages summarize the statements and options for the SGPLOT procedure.

The SGPLOT procedure produces 16 different types of plots that can be grouped into five general areas: basic X Y plots, band plots, fit and confidence plots, distribution graphs for continuous DATA, and distribution graphs for categorical DATA. The VECTOR statement is new with SAS 9.2 Phase 2; all the others are available with SAS 9.2 Phase 1. Many of these plot types can be used together in the same graph. In the preceding examples, we used the HISTOGRAM and DENSITY statements together to produce a histogram overlaid with a normal density curve. We also used three SERIES statements together to produce one graph with three different series lines. However, not all plot statements can be used with all other plot statements. Table 1 shows which statements can be used with which other statements. Table 1 also includes several options that can be used with many different plot statements.

Table 2 shows each of the 16 plot statements along with their basic syntax and selected options. The options listed in Table 2 are in addition to the options listed in Table 1.

In addition to the plot statements, there are some optional statements you might want to use. These statements can be used with any type of plot to control axes, or add reference lines and insets. Table 3 shows these statements with selected options.

Several types of plots can use the LINEATTR, MARKERATTR, or FILLATTR options to change the appearance of lines, markers, or fill (see Table 1). These options allow you choose values for the color of fill; the color, pattern and thickness of lines; and color, symbol, and size of markers. Table 4 gives the syntax for hard coding the values for these options. Note that it is also possible to use ODS styles to control these attributes, or to change them using the ODS Graphics Editor.

Even with all the options listed in these four tables, this is just a sample. Each plot statement has many possible options—we have listed only a few of the most useful. For a complete list of available options, see the SAS Help and Documentation for PROC SGPLOT.



Table 1. Compatibility of SGPLOT procedure statements and selected options. If a check mark appears in the box then the two statements or options can be used together.

	SCATTER	SERIES	STEP	NEEDLE	VECTOR	BAND	REG	LOESS	PBSPLINE	ELLIPSE	HBOX/VBOX	HISTOGRAM	DENSITY	HBAR/VBAR	HLINE/VLINE	DOT
<b>Basic X Y Plots</b>																
<i>PLOTNAME X=var Y=var / options;</i>																
SCATTER	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓						
SERIES	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓						
STEP	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓						
NEEDLE	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓						
VECTOR	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓			
<b>Band Plots</b>																
<i>BAND X=var UPPER=var LOWER=var / options; (You can also specify numeric values for upper and lower.)</i>																
BAND	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓						
<b>Fit and Confidence Plots</b>																
<i>PLOTNAME X=var Y=var / options;</i>																
REG	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓						
LOESS	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓						
PBSPLINE	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓						
ELLIPSE	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓						
<b>Distribution Graphs – Continuous DATA</b>																
<i>PLOTNAME response-var / options;</i>																
HBOX or VBOX																
HISTOGRAM					✓								✓			
DENSITY					✓							✓	✓			
<b>Distribution Graphs – Categorical DATA</b>																
<i>PLOTNAME category-var / options;</i>																
HBAR or VBAR														✓	✓	✓
HLINE or VLINE														✓	✓	✓
DOT														✓	✓	✓
<b>Selected Options in Plot Statements</b>																
GROUP <i>/GROUP=var;</i>	✓	✓	✓	✓	✓	✓	✓	✓	✓					✓	✓	✓
TRANSPARENCY <i>/TRANSPARENCY=value ;</i>	✓	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓	✓
MARKERS <i>/MARKERS;</i>		✓	✓	✓											✓	
NOMARKERS <i>/NOMARKERS;</i>							✓	✓	✓							
LEGENDLABEL <i>/LEGENDLABEL= 'text-string';</i>	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
FILLATTRS <i>/FILLATTRS= (attribute=val);</i>						✓				✓		✓		✓		
LINEATTRS <i>/LINEATTRS= (attribute=val);</i>		✓	✓	✓	✓	✓	✓	✓	✓	✓			✓		✓	
MARKERATTRS <i>/MARKERATTRS= (attribute=val);</i>	✓	✓	✓	✓			✓	✓	✓						✓	✓

Table 2. SGPLOT plot statements and selected options.

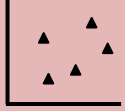

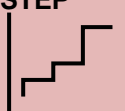


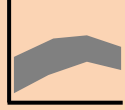
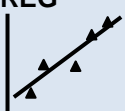

	SYNTAX	SELECTED OPTIONS	
<b>SCATTER</b> 	SCATTER X=var Y=var/options;	DATALABEL=var	Displays a label for each DATA point
<b>SERIES</b> 	SERIES X=var Y=var/options;	BREAK	Creates a break in the line for each missing value
		CURVELABEL	Labels the series curve using the Y variable label
<b>STEP</b> 	STEP X=var Y=var/options;	BREAK	Creates a break in the line for each missing value
		CURVELABEL	Labels the step curve using the Y variable label
<b>NEEDLE</b> 	NEEDLE X=var Y=var/options;	BASELINE=val	Specifies a numeric value on the Y axis for the baseline
<b>VECTOR</b> 	VECTOR X=var Y=var/options;  (Note: VECTOR is new with SAS 9.2. Phase 2.)	XORIGIN=val	Specifies X coordinate for origin, either numeric value or numeric variable.
		YORIGIN=val	Specifies Y coordinate for origin, either numeric value or numeric variable.
<b>BAND</b> 	BAND X=var UPPER=var LOWER=var/options;	FILL NOFILL	Specifies if fill is visible or not
		OUTLINE NOOUTLINE	Specifies if outline is visible or not
<b>REG</b> 	REG X=var Y=var/options;	ALPHA=val	Specifies confidence level (default 0.05)
		CLI	Displays confidence limits for individual predicted values
		CLM	Displays confidence limits for mean predicted values
<b>LOESS</b> 	LOESS X=var Y=var/options;	ALPHA=val	Specifies confidence level (default 0.05)
		CLM	Displays confidence limits
		INTERPOLATION=	Specifies degree of interpolation: CUBIC (default) or LINEAR

Table 2 (continued). SGPLOT plot statements and selected options.



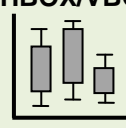
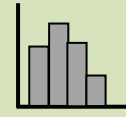

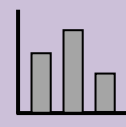
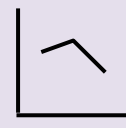
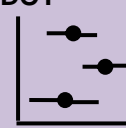
	SYNTAX	SELECTED OPTIONS	
	PBSPLINE X=var Y=var/ options;	ALPHA=val	Specifies confidence level (default 0.05)
		CLI	Displays confidence limits for individual predicted values
		CLM	Displays confidence limits for mean predicted values
	ELLIPSE X=var Y=var/options;	ALPHA=val	Specifies confidence level for the ellipse
		TYPE=	Specifies type of ellipse: MEAN or PREDICTED (default)
	VBOX response-var/options; HBOX response-var/options;	CATEGORY=var	A box plot is created for each value of the category variable
		MISSING	Creates box plot for missing values of category variable
	HISTOGRAM response-var/ options;	SHOWBINS	Places tic mark at midpoint of bin
		SCALE=	Specifies scale for vertical axis: PERCENT (default), COUNT or PROPORTION
	DENSITY response-var/ options;	SCALE=	Specifies scale for vertical axis: DENSITY (default), PERCENT COUNT or PROPORTION
		TYPE=	Specifies type of density function: NORMAL (default) or KERNEL
	VBAR category-var/options; HBAR category-var/options;	RESPONSE=var	Specifies a numeric response variable for plot
		STAT=	Specifies statistic for axis of response variable (if specified): MEAN, or SUM (default)
		BARWIDTH=num	Specifies numeric value for width of bars (default 0.8)
	VLINE category-var/options; HLINE category-var/options;	RESPONSE=var	Specifies numeric response variable for plot
		STAT=	Specifies statistic for axis of response variable (if specified): MEAN, or SUM (default)
	DOT category-var/options;	RESPONSE=var	Specifies numeric response variable for plot
		STAT=	Specifies statistic for axis of response variable (if specified): MEAN, or SUM (default)
		LIMITSTAT=	Specifies statistic for limit lines (must use STAT=MEAN): CLM (default), STDDEV, or STDERR

Table 3. Selected optional statements with selected options.

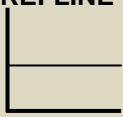
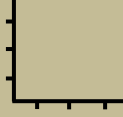




	SYNTAX	SELECTED OPTIONS	
<b>REFLINE</b> 	<code>REFLINE value1 value2 ... / options;</code>	<code>AXIS=</code>	Specifies axis for reference line: X, Y (default), X2, or Y2
		<code>LABEL=( )</code>	Creates labels for reference lines: ('text1' 'text2' ...)
		<code>LABELLOC=</code>	Specifies placement of label with respect to plot area: INSIDE (default) or OUTSIDE
		<code>LINEATTRS=( )</code>	Specifies attributes for reference line: ( <i>attribute=value</i> )
<b>XAXIS/YAXIS</b> 	<code>XAXIS options; YAXIS options;</code>	<code>GRID</code>	Creates grid line at each tick on the axis
		<code>LABEL='text'</code>	Specifies a label for the axis
		<code>TYPE=</code>	Specifies type of axis: DISCRETE, LINEAR, LOG, or TIME
		<code>VALUES=( )</code>	Specifies values for tics on the axis: ( <i>num1,num2,...</i> ) or ( <i>num1 TO num2 BY increment</i> )
<b>INSET</b> 	<code>INSET 'text1' 'text2' ... / options;</code>	<code>BORDER</code>	Creates a border around text box
		<code>POSITION=</code>	Specifies position of text box within plot area: BOTTOM, BOTTOMLEFT, BOTTOMRIGHT, LEFT, RIGHT, TOP, TOPLEFT, or TOPRIGHT

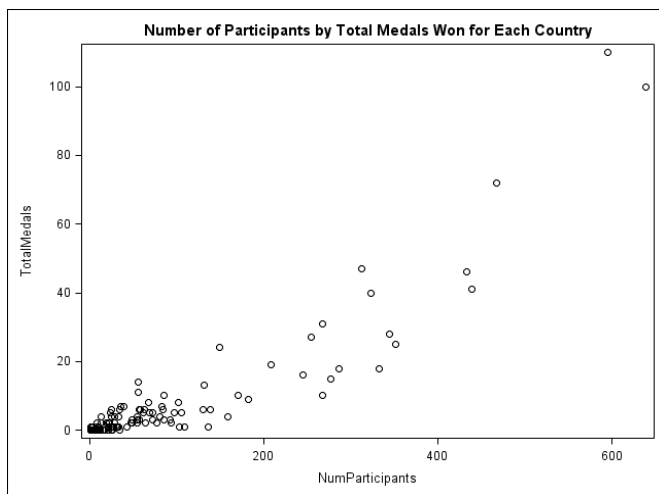
Table 4. The attribute options for plot statements.

	SYNTAX	ATTRIBUTES	
<b>FILLATTRS</b> 	<code>/FILLATTRS=(attribute=value);</code>	<code>COLOR=</code>	Specifies color for fill including: AQUA, BLACK, BLUE, FUCHSIA, GREEN, GRAY, LIME, MAROON, NAVY, OLIVE, PURPLE, RED, SILVER, TEAL, WHITE, and YELLOW
<b>LINEATTRS</b> 	<code>/LINEATTRS=(attribute=value);</code>	<code>COLOR=</code>	Specifies color for fill including: AQUA, BLACK, BLUE, FUCHSIA, GREEN, GRAY, LIME, MAROON, NAVY, OLIVE, PURPLE, RED, SILVER, TEAL, WHITE, and YELLOW
		<code>PATTERN=</code>	Specifies pattern for line including: SOLID, DASH, SHORTDASH, LONGDASH, DOT, DASHDASHDOT, and DASHDOTDOT
		<code>THICKNESS=val</code>	Specifies thickness of line. Value can include units: CM, IN, MM, PCT, PT, or PX (default)
<b>MARKERATTRS</b> 	<code>/MARKERATTRS=(attribute=value);</code>	<code>COLOR=</code>	Specifies color for fill including: AQUA, BLACK, BLUE, FUCHSIA, GREEN, GRAY, LIME, MAROON, NAVY, OLIVE, PURPLE, RED, SILVER, TEAL, WHITE, and YELLOW
		<code>SIZE=val</code>	Specifies size of marker. Value can include units: CM, IN, MM, PCT, PT, or PX (default)
		<code>SYMBOL=</code>	Specifies symbol for marker including: CIRCLE, CIRCLEFILLED, DIAMOND, DIAMONDFILLED, PLUS, SQUARE, SQUAREFILLED, STAR, STARFILLED, TRIANGLE, and TRIANGLEFILLED

## THE SGPNEL PROCEDURE

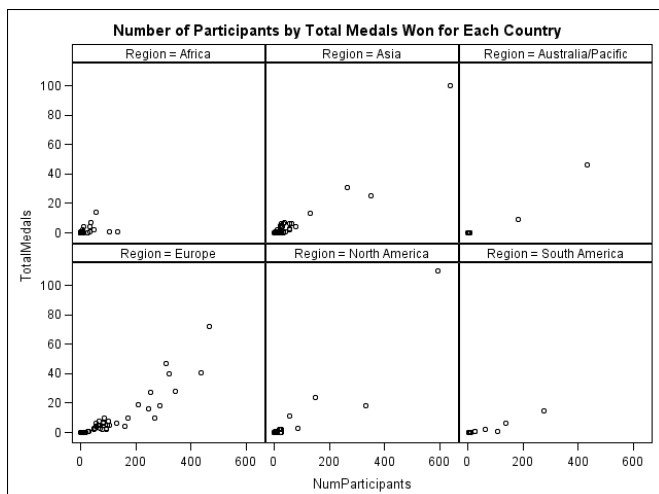
The SGPNEL procedure produces nearly all the same types of graphs as SGPNEL, but instead of displaying only one plot per image, SGPNEL can display several plots in a single image. A separate plot is produced for each level of the variable you specify in the PANELBY statement. To illustrate how SGPNEL works, here is the SGPNEL code to produce a scatter plot of the number of participants by the total medals won for each country.

```
*Scatter Plot using SGPNEL;
PROC SGPNEL DATA=Countries;
  SCATTER X = NumParticipants Y = TotalMedals;
  TITLE 'Number of Participants by Total Medals Won for Each Country';
RUN;
```



The syntax for SGPNEL is almost identical to SGPNEL, so it is easy to convert SGPNEL code to SGPNEL by making just a couple changes to your code. To produce a panel of plots, replace the SGPNEL keyword with SGPNEL, and add the PANELBY statement. The PANELBY statement must appear before any statements that create plots. Here, the PANELBY statement produces a separate plot for each value of REGION.

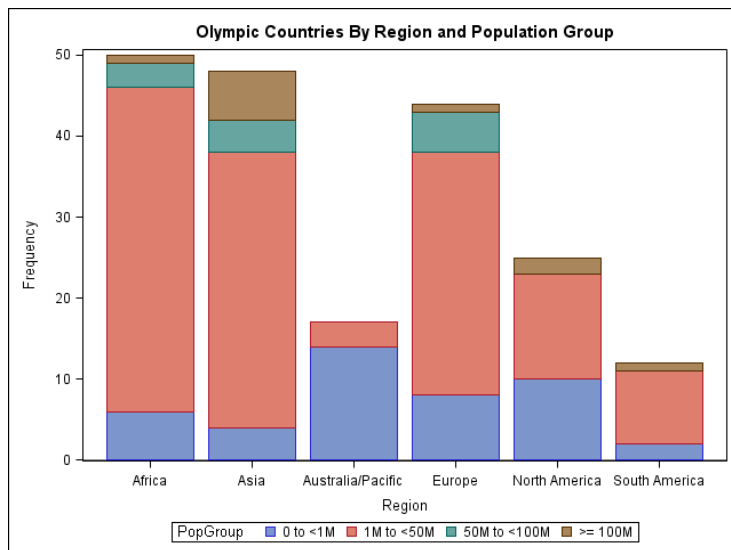
```
*Scatter plots using SGPNEL;
PROC SGPNEL DATA=Countries;
  PANELBY Region;
  SCATTER X = NumParticipants Y = TotalMedals;
  TITLE 'Number of Participants by Total Medals Won for Each Country';
RUN;
```



## CHANGING THE ODS STYLE

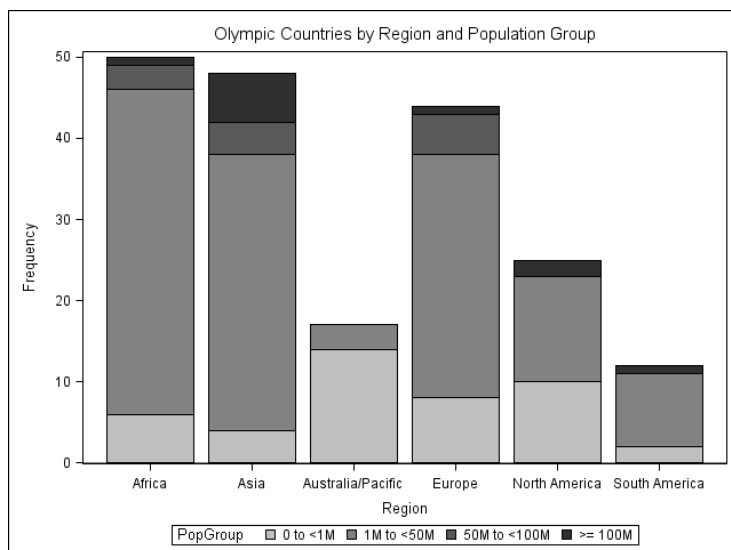
The following code creates a bar chart using the default ODS style template and destination. Since no destination is specified, the output will go to the LISTING destination. In the LISTING destination, tabular output is rendered as plain text so styles apply only to graphical output. The default style for the LISTING destination is named LISTING.

```
* Bar chart with default style;
PROC SGPLOT DATA = Countries;
  VBAR Region / GROUP = PopGroup;
  TITLE 'Olympic Countries by Region and Population Group';
RUN;
```



You can use the STYLE= option in an ODS destination statement to specify a style for your output including graphs. The following ODS LISTING statement changes the style to JOURNAL.

```
* Change ODS style template;
ODS LISTING STYLE = JOURNAL;
PROC SGPLOT DATA = Countries;
  VBAR Region / GROUP = PopGroup;
  TITLE 'Olympic Countries by Region and Population Group';
RUN;
```



You can use the same style templates for graphs as you do for tabular output. However, some styles are better suited for statistical graphs than others. The following table lists styles that are recommended for graphical results.

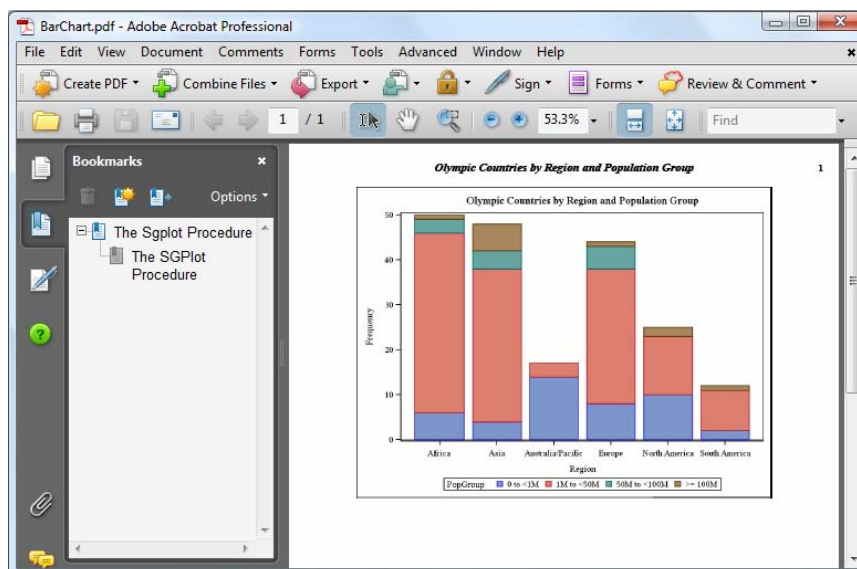
<u>Desired Output</u>	<u>Style Template</u>	<u>Default for Destination</u>
Color	ANALYSIS DEFAULT LISTING PRINTER RTF STATISTICAL	HTML LISTING (graphs only) PRINTER, PDF, PS RTF
Gray scale	JOURNAL	
Black and white	JOURNAL2	

Every destination has a default style associated with it. This means that if you change the destination for a graph, its appearance may change too.

### CHANGING THE ODS DESTINATION

You can send ODS Graphics output to any ODS destination, and you do it in the same way that you would for tabular output, using ODS statements for that destination. These statements send a bar chart to the PDF destination.

```
* Send graph to PDF destination;
ODS PDF FILE = 'c:\MyPDFFiles\BarChart.pdf';
PROC SGPLOT DATA = Countries;
  VBAR Region / GROUP = PopGroup;
  TITLE 'Olympic Countries by Region and Population Group';
RUN;
ODS PDF CLOSE;
```



### SAVING ODS GRAPHICS OUTPUT AND ACCESSING INDIVIDUAL GRAPHS

For most destinations (including RTF and PDF), graphs are integrated with tabular output into a single file. For these destinations, you can use the FILE= option to tell SAS where to save your output. This statement would create PDF output and save it in file named Report.pdf in a folder named MyPDFFiles.

```
ODS PDF FILE = 'c:\MyPDFFiles\Report.pdf';
```

However, when you send output to the LISTING or HTML destinations, graphs are saved in individual files separate from tabular output. By default these files are saved in the current SAS working directory. (In the SAS windowing environment, the path of the current SAS working directory appears in the lower-right corner of the SAS window.) For the LISTING and HTML destinations, you can use the GPATH= option to tell SAS where to save individual graphs. This statement would save all graphs sent to the HTML destination in a folder named MyGraphs.



```
ODS HTML GPATH = 'c:\MyGraphs' ;
```

For the HTML destination, these individual graph files are linked to the tabular output so that they appear integrated when viewed in a Web browser. Therefore, when you send graphical output to the HTML destination, you must be careful to save those files in directories where they can be found by anyone who displays the HTML file in a Web browser.

If you are writing a paper or creating a presentation, you may need to access individual graphs. For the HTML, PDF, and RTF destinations in the Windows operating environment, you can simply copy and paste images when you view them in SAS. You cannot copy and paste images from the LISTING destination, but since the individual graphs are saved in separate files, you can open or insert those files as you would any standard image file.

## SPECIFYING PROPERTIES OF IMAGES

Using the ODS GRAPHICS statement, you can control many aspects of your graphics files including the image format, name, height, and width. LISTING is the best destination for this since it offers the most image formats, and saves images in separate files. Here are a few of the options you can specify in the ODS GRAPHICS statement:

<u>Option</u>	<u>Description</u>	<u>Example</u>
IMAGEFMT=	specifies image format	IMAGEFMT = JPEG
IMAGENAME=	specifies base filename for image	IMAGENAME = 'NewGraph'
HEIGHT=	specifies height	HEIGHT = 4IN
WIDTH=	specifies width	WIDTH = 8CM
RESET	resets any preceding options to default values	RESET

Image formats supported for the LISTING destination include PNG (the default), BMP, GIF, JPEG, JPG, PDF, PS, TIFF, and many others.

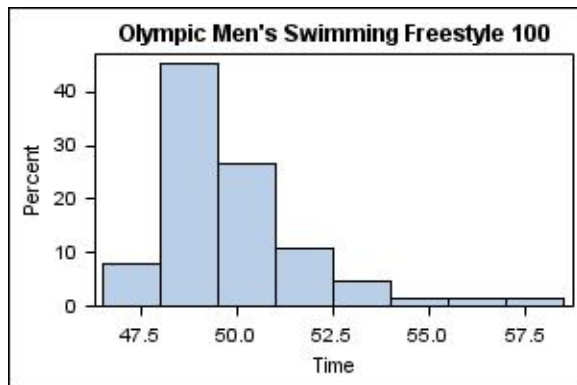
The default name for an image file is the name of its ODS output object. (You can use ODS TRACE statements to find the name of any ODS output object.) If you create more than one image file, SAS will append numerals to the end of the image name. For example, if you specified an image name of Olympic, then the files would be named Olympic, Olympic1, Olympic2, Olympic3, and so on. If you rerun your code, SAS will, by default, continue counting so that your graphics files will not be overwritten. If you want to start at zero each time you run your code, then specify the RESET option before the IMAGENAME= option.

In most cases, the default size for graphs is 640 pixels wide by 480 pixels high. If you specify only one dimension (width but not height, or vice versa), then SAS will adjust the other dimension to maintain a default aspect ratio of 4/3. You can specify width and height in these units: CM, IN, MM, PT, or PX.

The following code creates a graph in JPEG format that is 2 inches by 3 inches. The file will be named Final.jpeg and will be saved in a folder named MyGraphs.

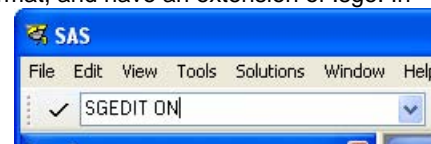
```
ODS GRAPHICS / RESET IMAGENAME = 'Final' IMAGEFMT =JPEG
  HEIGHT = 2in WIDTH = 3in;
ODS LISTING GPATH = 'c:\MyGraphs' ;

* Histograms;
PROC SGPLOT DATA = Freestyle;
  HISTOGRAM Time;
  TITLE "Olympic Men's Swimming Freestyle 100";
RUN;
```



### MODIFYING GRAPHS IN THE ODS GRAPHICS EDITOR

You can use the ODS Graphics Editor to make one-time changes to graphs. The ODS Graphics Editor is available for all operating environments except z/OS and VMI. Before you can edit graphs, you must first tell SAS to create graphs in an editable form. Editable graphs are saved in ODS Graphics Editor file format, and have an extension of .sge. In SAS 9.2 Phase 1 and later, you can create editable graphs by clicking the Results window, and then submitting the **SGEDIT ON** command in the command box of the SAS windowing environment. If you want to turn off the creation of editable graphs (to save resources, for example), then submit the command, **SGEDIT OFF**.



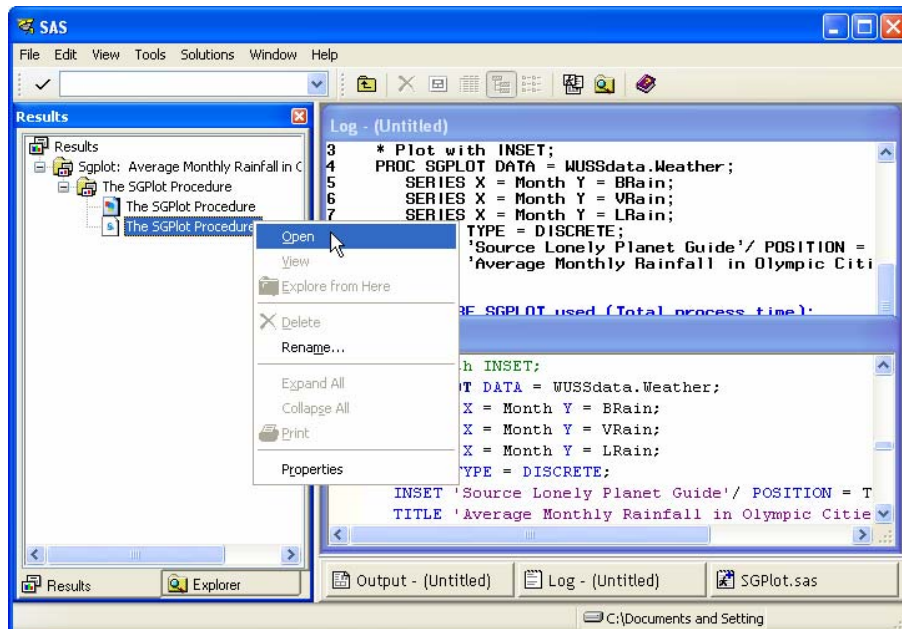
If you are using SAS 9.2 Phase 2, you can also create editable graphs in your program by adding this statement.

```
ODS LISTING SGE = ON;
```

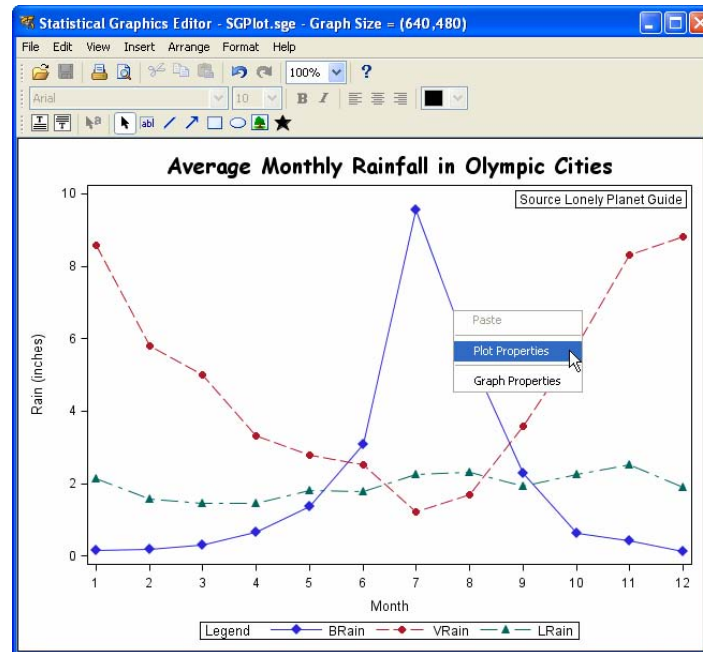
After turning on editable graphs, then submit the code that creates the graphs. If you want to stop creating editable graphs, then submit this statement

```
ODS LISTING SGE = OFF;
```

To open a graph in the ODS Graphics Editor in SAS 9.2 Phase 1, right-click the icon for the PNG image in the Results window, and select **Edit** from the pop-up menu. To open a graph in the ODS Graphics Editor in SAS 9.2 Phase 2, right-click the icon for the SGE image in the Results window, and select **Open** from the pop-up menu.



You can make many changes in the ODS Graphics Editor including changing the overall style, adding and editing titles and axis labels, and changing line types and markers. To make changes, use the menus and toolbar, or right-click individual parts of your plot (such as lines, titles, and axis labels) to open pop-up menus for that feature.



When you are done, you can use File ► Save As to save your graph in either PNG or SGE format. If you save it in SGE format, then you can return to edit the file again at a later time.

## DOWNLOADING THE ODS GRAPHICS EDITOR

The ODS Graphics Editor is included with SAS/GRAPH software. Sometimes, however, the editor is not installed. If that is the case, then when you try to open an SGE image, you will see this message

This file does not have a program associated with it for performing this action. Create an association in the Folder Options control panel.

If you see this message, then you need to install the editor. For the Windows or Linux operating environments, you can download the editor from the Downloads and Hot Fixes for SAS/GRAPH page on support.sas.com. The link is

<http://www.sas.com/apps/demosdownloads/setupcat.jsp?cat=SAS%2FGRAPH+Software>

## CONCLUSIONS

ODS Graphics and the SGPLOT procedure introduce an exciting new way of producing high quality graphs using SAS. While PROC SGPLOT doesn't completely replace traditional SAS/GRAPH procedures, it does offer a wide variety of graphs using simple syntax. Because it is part of the Output Delivery System, you can use the same styles and destinations that you use for tabular output. A bonus of learning SGPLOT is that you can easily produce a panel of plots by converting your SGPLOT code to SGPANEL.

## REFERENCES

Central Intelligence Agency (2007). "The World Factbook." <http://www.cia.gov/cia/publications/factbook/index.html>.

Delwiche, L. D. Slaughter, S. J. (2008). *The Little SAS Book: A Primer, Fourth Edition*. SAS Institute, Cary, NC.

SAS Institute Inc. (2008). "SAS/GRAPH 9.2: Statistical Graphics Procedures Guide." <http://support.sas.com/documentation/onlinedoc/graph/index.html>.

**ABOUT THE AUTHORS**

Lora Delwiche and Susan Slaughter are the authors of *The Little SAS Book: A Primer*, and *The Little SAS Book for Enterprise Guide* which are published by SAS Institute. The authors may be contacted at:

Susan J. Slaughter  
(530)756-8434  
susan@avocetsolutions.com

Lora D. Delwiche  
(530) 752-9321  
lldelwiche@ucdavis.edu

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.