

## Paper 235-2012

## Comparison of SAS® Graphic Alternatives, Old and New

LeRoy Bessler, Bessler Consulting and Research, Mequon, Milwaukee, WI

### ABSTRACT

Since my first encounter with SAS/GRAPH® software in 1981, I have tried to create communication-effective SAS graphics, rather than simply accept the defaults. This tutorial shows alternative customized SAS Version 9.3 graphic solutions, using both the traditional G procedures and the new SG (Statistical Graphics) procedures.

It covers graphs written to disk to later be inserted into Microsoft PowerPoint, Word, or Excel, or simply printed, but also demonstrates the benefits of web-enabled SAS graphs, as well as web-enabled SAS graphs linked forward and backward with an Excel spreadsheet of the data graphed. A graph enables quick and easy inference, but the actual data assures correct inference. A spreadsheet of data can be post-processed however your information recipient wishes, using a tool available and familiar to almost anyone.

### INTRODUCTION

My focus in this paper is on graphs used for management reporting in business, government, or other organizational settings, not statistical or heavy-duty analytical graphs. Although the SG (Statistical Graphics) procedures were originally developed for statistical graphics, they can be used for management reporting. By graphs for management reporting I mean graphs or web graphs used to answer several common questions. How are things going? Are they better, worse, or about the same? How do measurements for different entities compare? Questions like these are commonly answered visually with time plots (or time series graphs), bar charts, and pie charts.

Though it is not the avowed purpose of this tutorial, I will address visual communication effectiveness along the way, especially in the slide presentation. Attendees at the tutorial will receive a copy of my modestly (OK, grandiosely) titled "*Principia Graphika*" as a takeaway. Any way that a tool adversely affects communication effectiveness will be pointed out. *Principia Graphika* is not SAS-version-focused, and it applies to any visual communication tool.

My scope does include static graphs, i.e., ones destined directly for print (probably rare these days) or for imbedding in PowerPoint for presentations, in PDF for Adobe Acrobat Reader, or in RTF for Microsoft Word. However, web-enabled graphs are my preferred choice now. Web enablement provides quick and easy navigation between graph and graph or between graph and table. It also supports what is officially called ALT text, but often is instead referred to as pop-up text (at least by me), flyover text, mouseover text, hover text, floatover text, or tool tips.

ALT text is an accessibility aid. For visually impaired web users, the HTML source code to support ALT text can be converted into audio by so-called screen reader software. But ALT text can be useful for any web user. A problem with many graphs is that you can not necessarily hard-annotate everything that you would like to communicate about a plot point, bar, pie slice, geographic unit area, or whatever. In any case, it is preferable to guessing the values for a plot point by comparing its location with axis tick marks. ALT text can be used for anything you want, as much as you want, formatted with line breaks wherever you want to enhance readability. You can even format it as a little table:

Your Label for Yvar = y value formatted however you like  
Your Label for Xvar = x value formatted however you like  
Any comments that you might have

For ways to supplement traditional graphic communication, see my 2006 paper (Reference 1) Multi-Media Wizardry: How To Make ODS Outputs That "Dance and Sing" or, here at SAS Global Forum 2012, the SAS staff presentation (Reference 2) on Using SAS/GRAPH to Create Visualizations That Also Support Tactile and Auditory Interaction.

Among the options that web enabled graphs offer is the ability to connect them, forwards and backwards, with SAS-created spreadsheets. Besides allowing detail look-up, the benefit of a spreadsheet rather than just a SAS table is that the user then has the option of post-processing the graph-supporting data in any way desired, using a tool that almost any user already has and knows how to use. Though my Abstract contains a commitment to cover this topic here, page count limits dictate that I must refer you to my prior work in Reference 3. I regret the inconvenience.

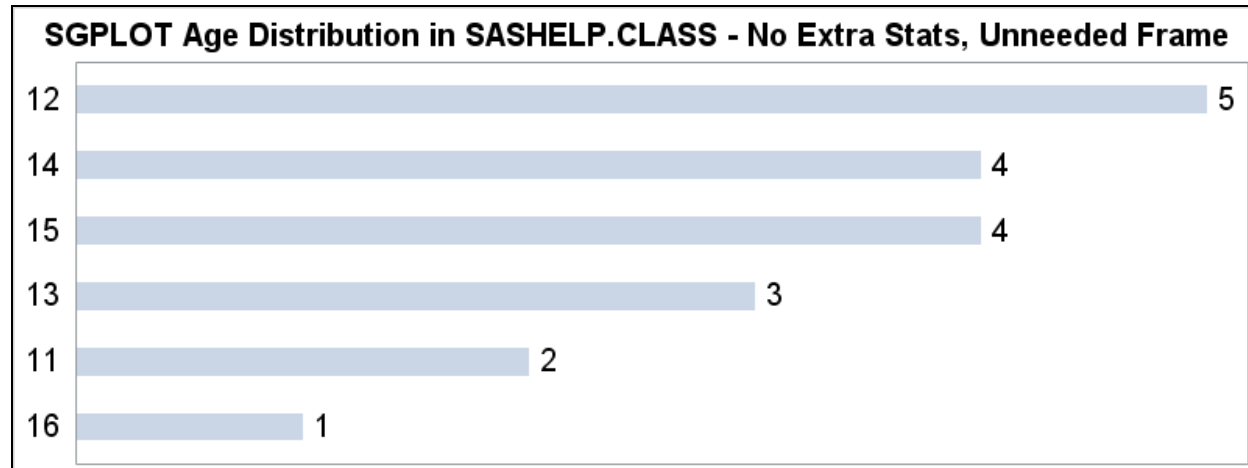
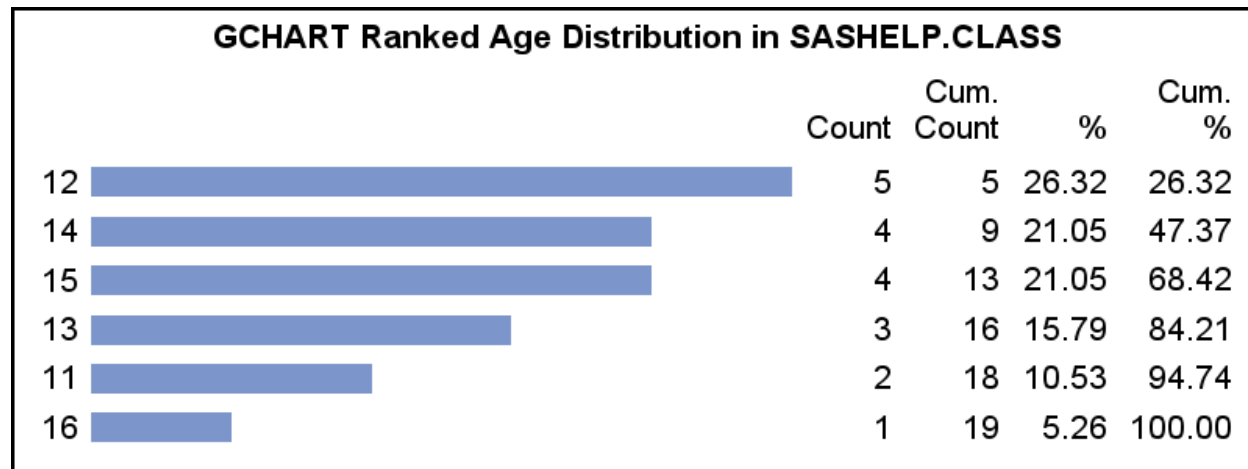
Bar charts and pie charts can be created with all detail in the image area. The code here routes the charts to a disk location, from which they can be inserted in a slide or a Microsoft Word document, but they could be web-enabled.

## BAR CHARTS

I have long been an advocate for horizontal bar charts rather than vertical bar charts. Vertical bar charts work well only when the bar labels are short. Tilted, or worse, vertical labels for vertical bars are somewhere between inelegant and outright anti-communicative. With V9.2, the length on labels was extended to 256, which is always adequate, and, for horizontal bar charts, always useful for longer labels that are, in fact, often needed. 256 would be impractical (no space left for the bar—unless you expand the image width, which IS possible), but it's a welcome, friendly limit.

Below is a comparison of GCHART and SGPLOT capabilities for a frequency distribution. Bar colors are defaults.

**NOTE:** One of my longstanding graphic design guidelines is: Show Them What's Important. Ordering the information by magnitude is an effective way to do that.



With SGPLOT, it is advantageous to have the frequency count automatically appended to the bar end without having to use ANNOTATE, but it is impossible to present the extra statistics available automatically in GCHART and impossible to suppress the useless frame around the bar display area.

Here is the code for the two charts above:

```
proc catalog cat=work.gseg kill; run; quit;
ods graphics off;
ods _all_ close;
ods listing;
options reset=all;
options gsfname=anyname gsfname=replace device=png ypixels=300 htext=16pt border;
filename anyname "C:\Graph Alternatives Old and New\GCHARTbarFreqChart.png";
```

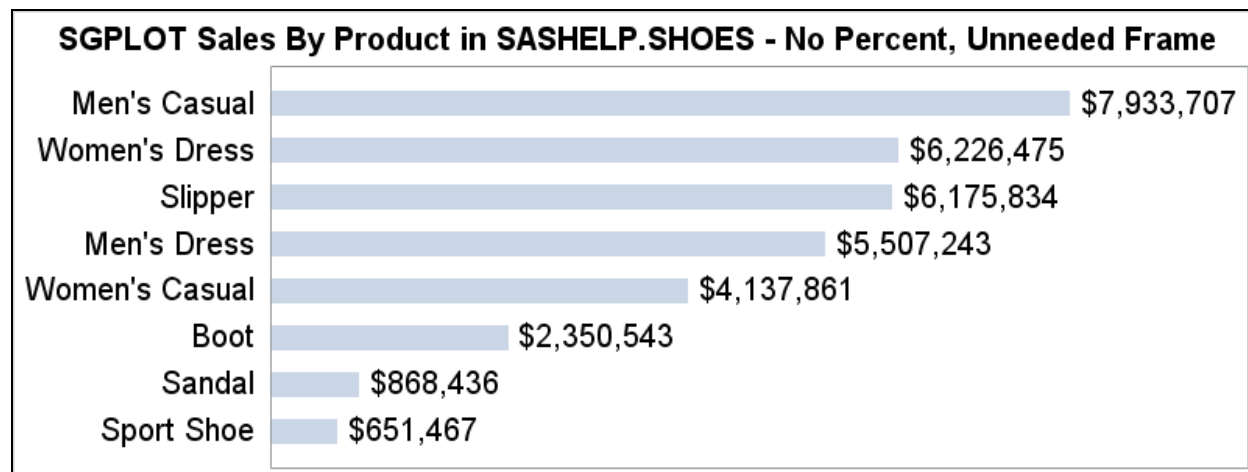
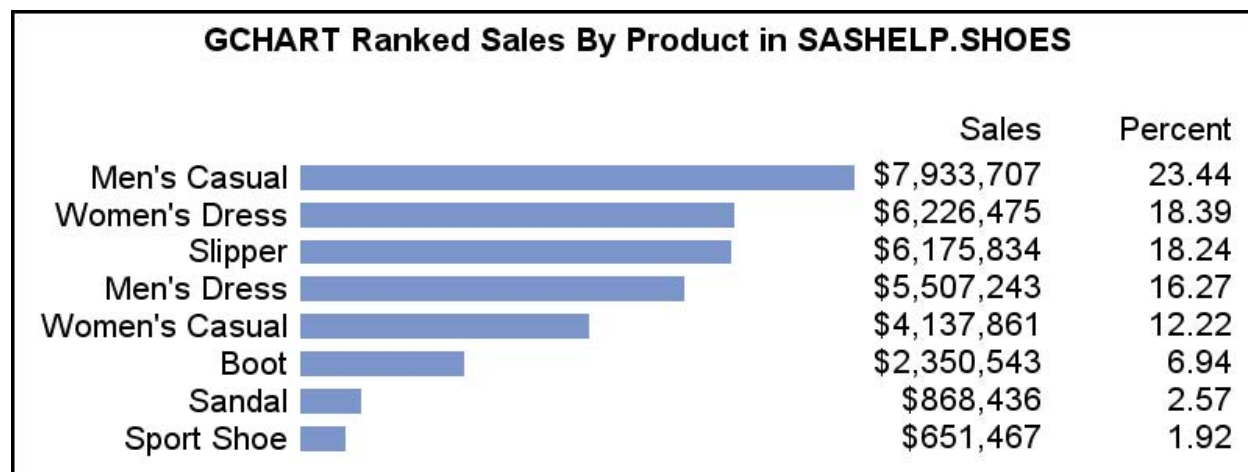
```

title height=1 ' ' justify=center height=16pt
      "GCHART Age Distribution in SASHELP.CLASS by Decreasing Frequency";
proc gchart data=sashelp.class;
axis1 label=none style=0;
axis2 label=none style=0 value=none major=none minor=none;
hbar Age / name='GCHARThbarFreqChart' maxis=axis1 raxis=axis2 width=3 space=2.25
      discrete descending noframe coutline=same
      freqlabel='Count' percentlabel='%' cfreqlabel='Cum. Count' cpercentlabel='Cum. %';
run; quit; title;

ods listing gpath="C:\Graph Alternatives Old and New";
ods graphics on / reset=all border=on height=300px width=800px
      imagename='SGPLOTThbarFreqChart';
title height=16pt
      "SGPLOT Age Distribution in SASHELP.CLASS - No Extra Stats, Unneeded Frame";
proc sgplot data=sashelp.class;
yaxis display=(nolabel noline noticks) valueattrs=(size=16pt);
xaxis display=none;
hbar Age / stat=freq categoryorder=respdesc datalabel datalabelATTRS=(SIZE=16pt)
      barwidth=0.4 nooutline;
run; quit; title;

```

Below is a comparison of GCHART and SGLOT capabilities for a summary bar chart. Bar colors are defaults.



With SGLOT, it is advantageous to have the sum value automatically appended to the bar end without having to use ANNOTATE, but it is impossible to present the useful percent easily available in GCHART and impossible to suppress the useless frame around the bar display area.

Here is the code for the two charts above:

```
proc catalog cat=work.gseg kill; run; quit;
ods graphics off; ods _all_ close; ods listing;
options reset=all device=png ypixels=300 htext=16pt border;
options gsfname=anyname gsfname=replace;
filename anyname "C:\Graph Alternatives Old and New\GCHARThbarSummaryChart.png";
title height=1 ' ' justify=center height=16pt
      "GCHART Ranked Sales By Product in SASHELP.SHOES";
proc gchart data=sashelp.shoes;
axis1 label=none style=0 value=(justify=right);
axis2 label=none style=0 value=none major=none minor=none;
hbar Product / sumvar=Sales name='GCHARThbarSummaryChart'
      maxis=axis1 raxis=axis2 width=2.5 space=1.5
      discrete descending noframe coutline=same
      sum sumlabel='Sales' percentsum pctsumlabel='Percent';
run; quit; title;

ods listing gpath="C:\Graph Alternatives Old and New";
ods graphics on / reset=all border=on height=300px width=800px
      imagename='SGPLOTthbarSummaryChart';
title height=16pt
      "SGPLOT Sales By Product in SASHELP.SHOES - No Percent, Unneeded Frame";
proc sgplot data=sashelp.shoes;
hbar Product / response=Sales stat=sum categoryorder=respdesc
      datalabel datalabelATTRS=(SIZE=16pt) barwidth=0.5 nooutline;
yaxis display=(nolabel noline noticks) valueattrs=(size=16pt);
xaxis display=none;
run; quit; title;
```

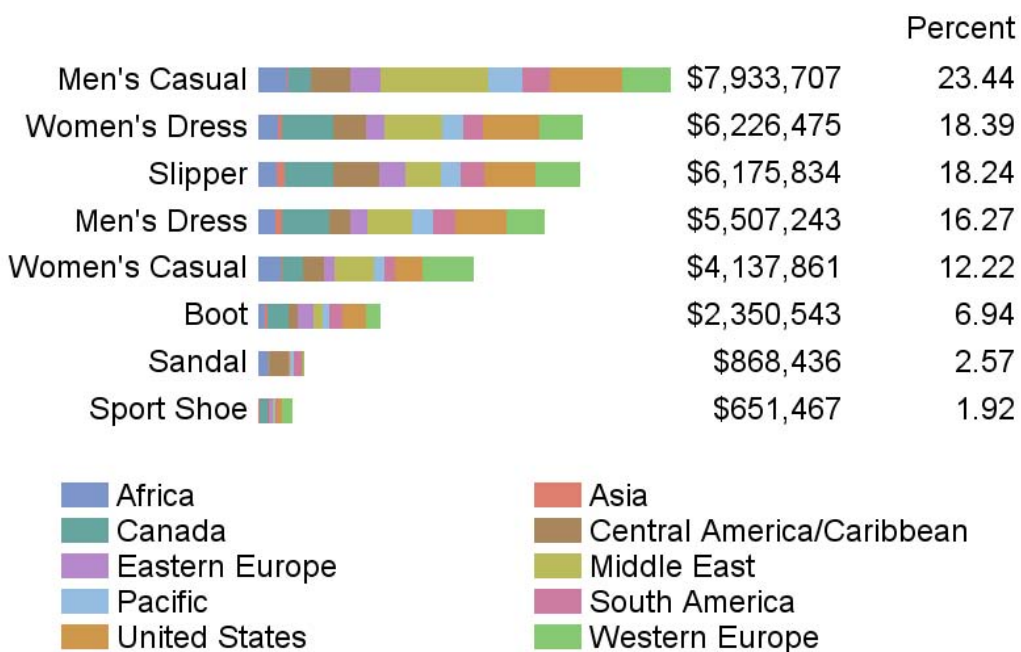
Here is the code for the two charts on the next page that require a legend:

```
proc catalog cat=work.gseg kill; run; quit;
ods graphics off; ods _all_ close; ods listing;
options reset=all device=png xpixels=800 ypixels=600 htext=2 border;
options gsfname=anyname gsfname=replace;
filename anyname
      "C:\Graph Alternatives Old and New\GCHARThbarSubGroupedSummaryChart.png";
title height=1 ' ' justify=center height=17pt
      "GCHART Sales By Region Within Ranked Product in SASHELP.SHOES";
proc gchart data=sashelp.shoes;
legend1 label=none shape=bar(5,1.5);
axis1 label=none style=0 value=(justify=right);
axis2 label=none style=0 value=none major=none minor=none;
hbar Product / sumvar=Sales name='GCHARThbarSubGroupedSummaryChart' subgroup=Region
      discrete descending noframe coutline=same
      maxis=axis1 raxis=axis2 width=1.5 space=1.5
      sum sumlabel=' ' percentsum pctsumlabel='Percent' legend=legend1;
run; quit; title;

ods listing gpath="C:\Graph Alternatives Old and New";
ods graphics on / reset=all border=on height=600px width=800px
      imagename='SGPLOTthbarWithGroupVar';
title1 height=16pt "SGPLOT Sales by Region within Product in SASHELP.SHOES";
title2 height=16pt color=red "No ranking is possible.";
title3 height=16pt color=red "Legend color samples are too small.";
title4 height=16pt color=red "MAXLEGENDAREA option is No Help, and";
title5 height=16pt color=red "reducing text height and bar width is No Help.";
proc sgplot data=sashelp.shoes;
yaxis display=(nolabel noline noticks) valueattrs=(size=16pt);
xaxis display=none;
hbar Product / response=Sales group=region
      barwidth=0.5 nooutline datalabel datalabelATTRS=(SIZE=16pt);
run; quit; title;
```

This GCHART result is better (ranked and with an easily color-distinguishable legend), but neither result allows you to know exact sales dollars by Region unless the chart is web-enabled with ALT text or linked to a spreadsheet or table. For SGPLOT, the categoryorder parameter is ignored, and the grouporder parameter does nothing.

### GCHART Sales By Region Within Ranked Product in SASHELP.SHOES



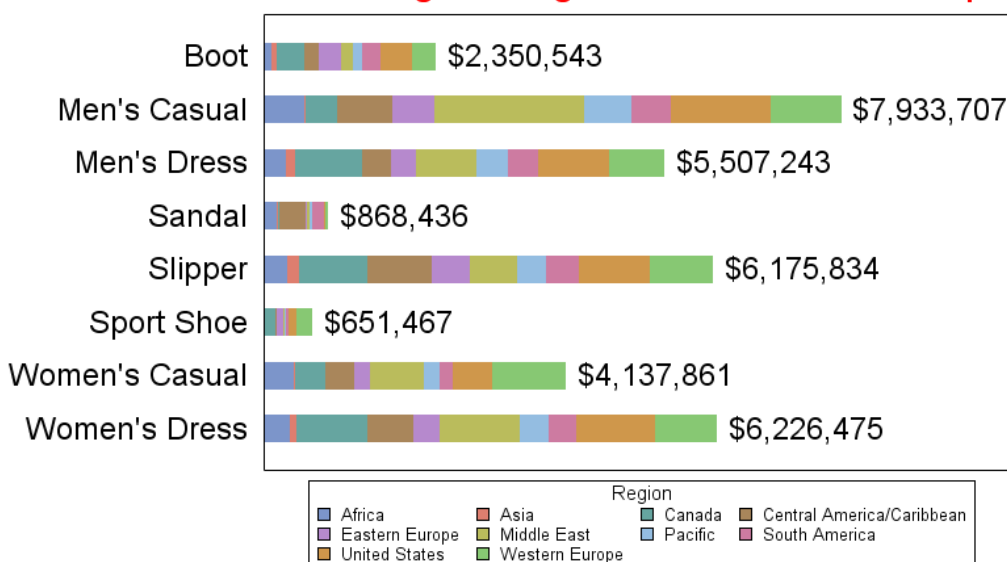
### SGPLOT Sales by Region within Product in SASHELP.SHOES

**No rank order possible.**

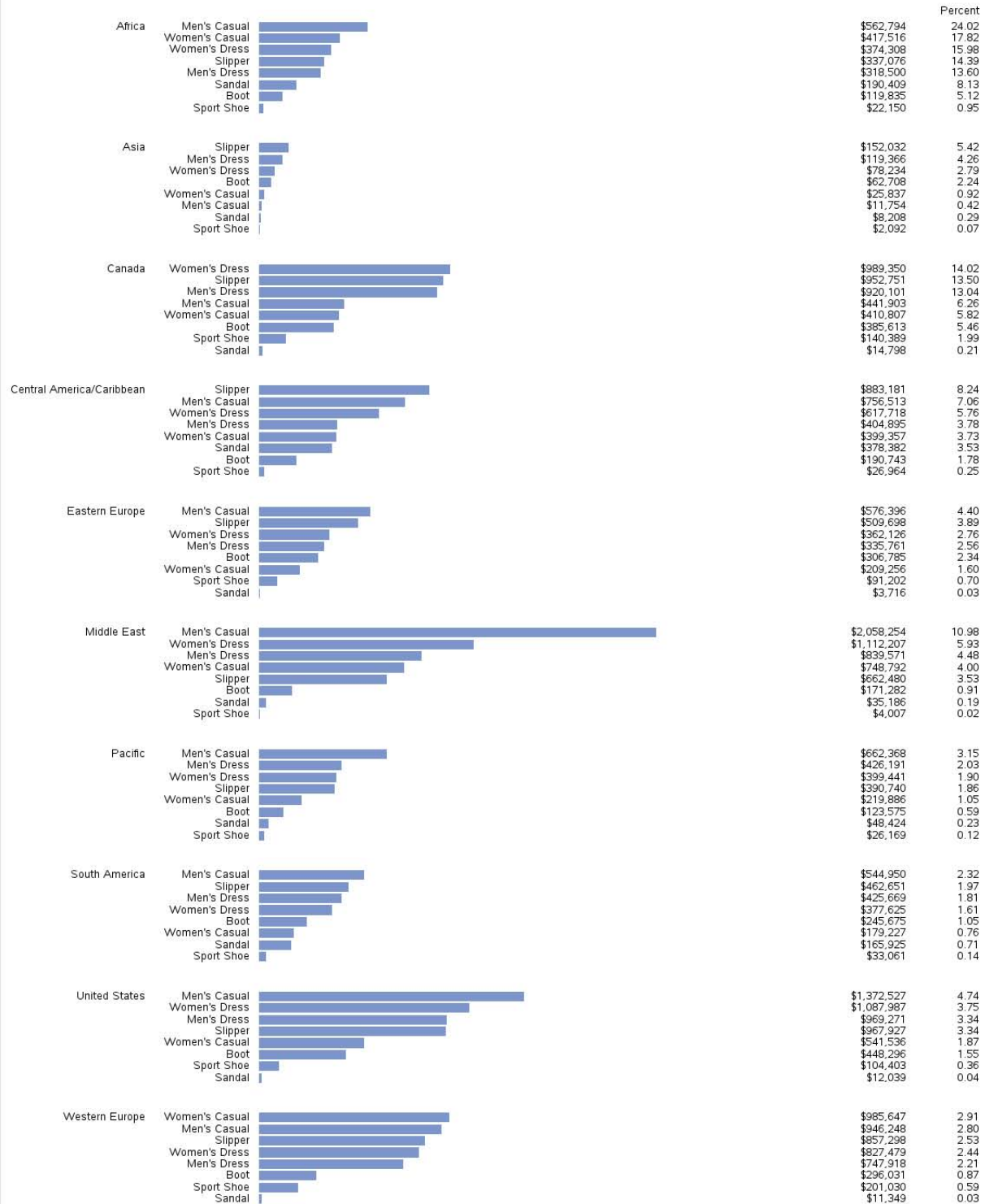
**Legend color samples are too small.**

**MAXLEGENDAREA option is No Help.**

**Reducing text height and bar width is No Help.**

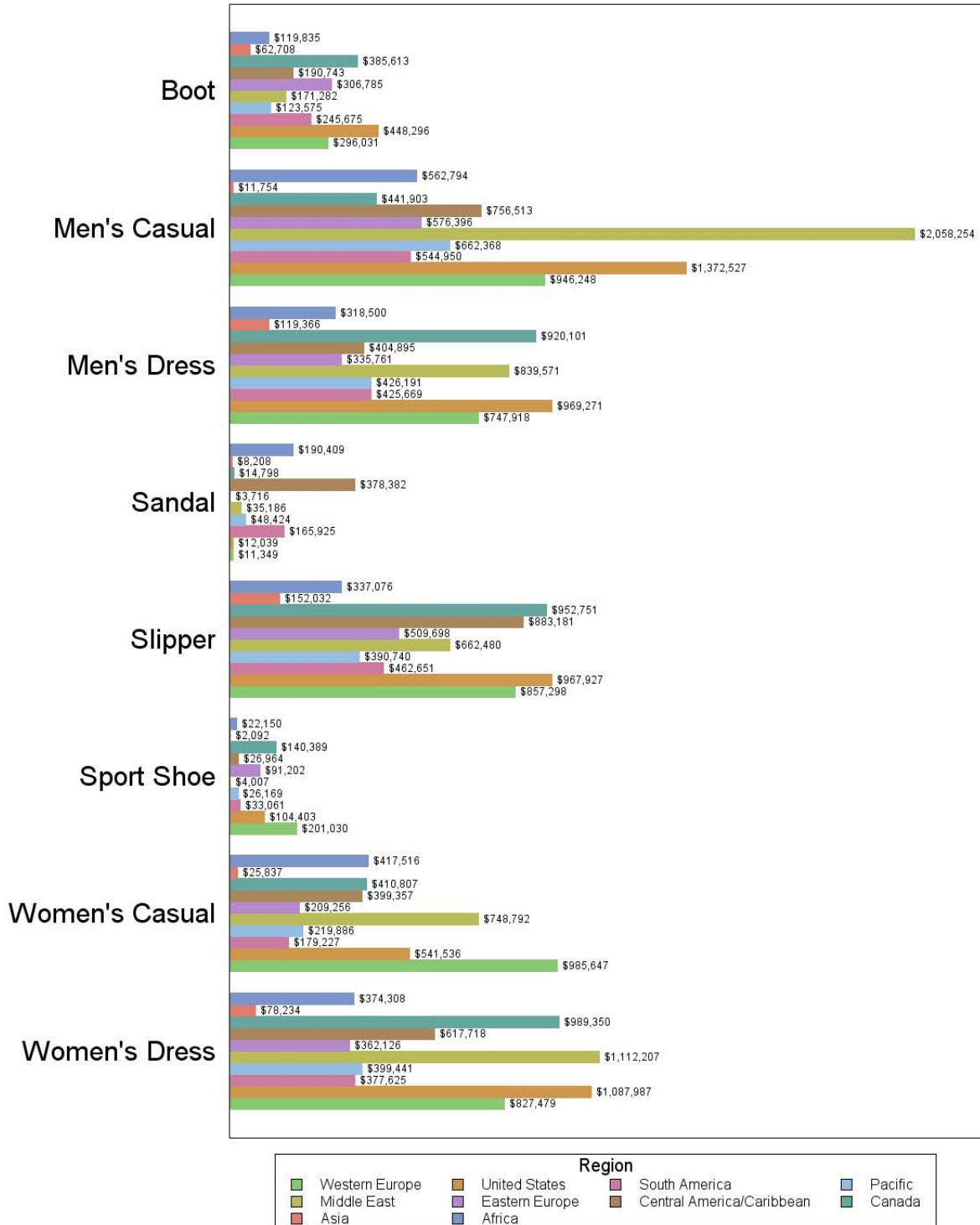


GCHART Sales by Ranked Product within Region in SASHELP.SHOES  
Sales Amount and Percent could be appended to Product for fully informative labels for longer bars, and remote columns at right could be eliminated. See example of preprocessing used for pie chart.





**SGPLOT Sales by Region within Product**  
**This is better than the SGPLOT stacked bar chart**  
**but there is no way to enlarge legend color samples**  
**or to rank Region bars within Product**



**SGPANEL Sales by Product within Region in SASHELP.SHOES**  
**Shared row header for columns makes CATEGORYORDER unusable.**  
**No control available for size of panel header text.**

	Africa	Asia
Boot	\$119,835	\$62,708
Men's Casual	\$562,794	\$11,754
Men's Dress	\$318,500	\$119,366
Sandal	\$190,409	\$8,208
Slipper	\$337,076	\$152,032
Sport Shoe	\$22,150	\$2,092
Women's Casual	\$417,516	\$25,837
Women's Dress	\$374,308	\$78,234
	Canada	Central America/Caribbean
Boot	\$385,613	\$190,743
Men's Casual	\$441,903	\$756,513
Men's Dress	\$920,101	\$404,895
Sandal	\$14,798	\$378,382
Slipper	\$952,751	\$883,181
Sport Shoe	\$140,389	\$26,964
Women's Casual	\$410,807	\$399,357
Women's Dress	\$989,350	\$617,718
	Eastern Europe	Middle East
Boot	\$306,785	\$171,282
Men's Casual	\$576,396	\$2,058,254
Men's Dress	\$335,761	\$839,571
Sandal	\$3,716	\$35,186
Slipper	\$509,698	\$662,480
Sport Shoe	\$91,202	\$4,007
Women's Casual	\$209,256	\$748,792
Women's Dress	\$362,126	\$1,112,207
	Pacific	South America
Boot	\$123,575	\$245,675
Men's Casual	\$662,368	\$544,950
Men's Dress	\$426,191	\$425,669
Sandal	\$48,424	\$165,925
Slipper	\$390,740	\$462,651
Sport Shoe	\$26,169	\$33,061
Women's Casual	\$219,886	\$179,227
Women's Dress	\$399,441	\$377,625
	United States	Western Europe
Boot	\$448,296	\$296,031
Men's Casual	\$1,372,527	\$946,248
Men's Dress	\$969,271	\$747,918
Sandal	\$12,039	\$11,349
Slipper	\$967,927	\$857,298
Sport Shoe	\$104,403	\$201,030
Women's Casual	\$541,536	\$985,647
Women's Dress	\$1,087,987	\$827,479



Here is the code used to create the charts on the three previous pages:

```
proc catalog cat=work.gseg kill; run; quit;
ods graphics off; ods _all_ close; ods listing;
goptions reset=all device=png xpixels=1155 ypixels=1600 border
    ftext='Albany AMT' htext=10pt;
goptions gsfname=anyname gsfmode=replace;
filename anyname
    "C:\Graph Alternatives Old and New\GCHARThbarGroupedSummaryChart.png";
title height=0.5 ' ' justify=center height=20pt
    "GCHART Sales by Ranked Product within Region in SASHELP.SHOES"
    justify=center height=18pt
    "Sales Amount and Percent could be appended to Product for fully informative labels
for longer bars," justify=center "and remote columns at right could be eliminated. See
example of preprocessing used for pie chart.";
proc gchart data=sashelp.shoes;
axis1 label=none style=0 value=(justify=right);
axis2 label=none style=0 value=none major=none minor=none;
axis3 label=none value=(justify=right);
hbar Product / sumvar=Sales name='GCHARThbarGroupedSummaryChart' group=Region
    discrete descending noframe coutline=same maxis=axis1 raxis=axis2 gaxis=axis3
    sum sumlabel=' ' percentsum pctsumlabel='Percent';
run; quit; title;

ods listing gpath="C:\Graph Alternatives Old and New";
ods graphics on / reset=all border=on height=1600px width=1155px
    imagename='SGPLOTThbarWithClusteredGroupVar';
title1 height=16pt "SGPLOT Sales by Region within Product";
title2 height=16pt color=blue "This is better than the SGLOT stacked bar chart";
title3 height=16pt color=red "but there is no way to enlarge legend color samples";
title4 height=16pt color=black "or to rank Region bars within Product";
proc sgplot data=sashelp.shoes;
yaxis display=(nolabel noline noticks) valueattrs=(size=16pt);
xaxis display=none;
hbar Product / response=Sales group=region groupdisplay=cluster
    nooutline datalabel datalabelATTRS=(SIZE=7pt);
run; quit; title;

ods listing gpath="C:\Graph Alternatives Old and New";
ods graphics on / reset=all border=on height=1600px width=1155px
    imagename='SGPANELhbar';
title1 height=16pt "SGPANEL Sales by Product within Region in SASHELP.SHOES";
title2 height=16pt "Shared row header for columns makes CATEGORYORDER unusable.";
title3 height=16pt "No control available for size of panel header text.";
proc sgpanel data=sashelp.shoes;
panelby region / novarname rows=5 columns=2;
rowaxis display=(nolabel noline noticks) valueattrs=(size=12pt);
colaxis display=none;
hbar Product / response=Sales barwidth=0.5 nooutline
    datalabel datalabelattrs=(SIZE=12pt);
run; quit; title;
```

## HOW TO SUPPLY PERCENT FOR AN SGLOT ORDERED HBAR SUMMARY CHART

To get Percent, it is necessary to imbed the percent in customized hbar labels. Here is the code:

```
proc summary data=sashelp.shoes nway;
class Product;
var Sales;
output out=ToPrep sum=SalesByProduct;
run;
```

```

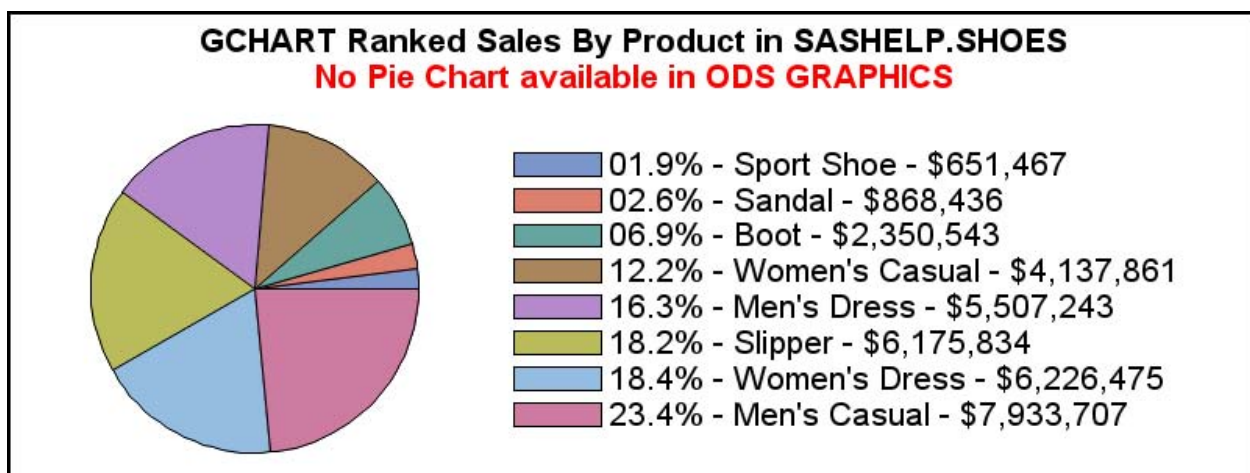
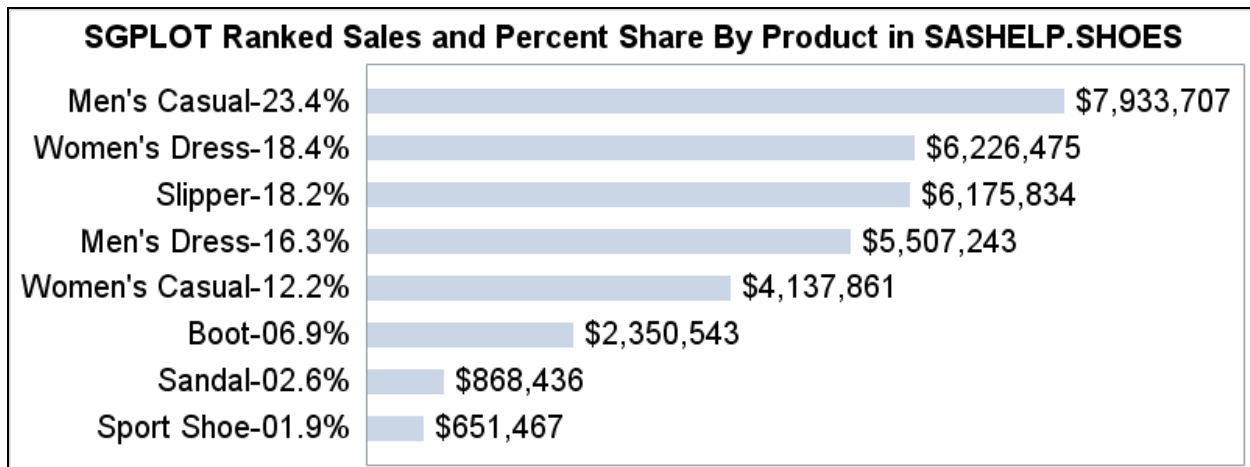
proc sql noprint;
select sum(SalesByProduct) into :TotalSales from ToPrep;
quit;

data ToChart;
length RankedProductWithPctShare $ 32; /* over-sized, but that is harmless */
set ToPrep;
ProductWithPctShare = trim(left(Product))
  || '-' || trim(left(put(((SalesByProduct / &TotalSales) * 100),z4.1))) || '%';
run;

ods listing gpath="D:\Graph Alternatives Old and New";
ods graphics on / reset=all border=on height=300px width=800px
  imagename='SGPLOTbbarSummaryChartWithPercent';
title height=16pt "SGPLOT Ranked Sales and Percent Share By Product in SASHELP.SHOES";
proc sgplot data=ToChart;
hbar ProductWithPctShare / response=SalesByProduct categoryorder=respdesc
  datalabel datalabelATTRS=(SIZE=16pt) barwidth=0.5 nooutline;
yaxis display=(nolabel noline noticks) valueattrs=(size=16pt);
xaxis display=none;
run; quit; title;

```

Here is the result:



Here is the code for the pie chart on the previous page:

```
proc summary data=sashelp.shoes nway;
class Product;
var Sales;
output out=ToPrep sum=SalesByProduct;
run;
proc sql noprint;
select sum(SalesByProduct) into :TotalSales from ToPrep;
quit;
data ToChart;
length SliceNameWithPercentAndValue $ 64; /* over-sized, but that is harmless */
set ToPrep;
SliceNameWithPercentAndValue =
  trim(left(put(((SalesByProduct / &TotalSales) * 100),z4.1))) || '% - ' ||
  trim(left(Product)) || ' - ' || trim(left(put(SalesByProduct,dollar10.)));
run;

proc catalog cat=work.gseg kill; run; quit;
ods graphics off;
ods _all_ close;
ods listing;
goptions reset=all device=png ypixels=300 htext=16pt border;
goptions gsfname=anyname gsfmode=replace;
filename anyname "C:\Graph Alternatives Old and New\GCHARTpieChart.png";
title height=1 ' ' justify=center height=16pt
  "GCHART Ranked Sales By Product in SASHELP.SHOES"
  justify=center color=red "No Pie Chart available in ODS GRAPHICS" ;
proc gchart data=ToChart;
legend1 label=none shape=bar(8,2.5) across=1
  position=(middle right outside) offset=(-5 PCT,0);
pie SliceNameWithPercentAndValue / sumvar=SalesByProduct name='GCHARTpieChart'
  discrete ascending coutline=black woutline=1 noheading legend=legend1
  slice=none value=none percent=none other=0 radius=35;
run; quit; title;
```

I prefer the pie legend to be ordered, top to bottom, from biggest slice to smallest slice. To get that, and to gain control of which slice color is associated with which product (not an obvious need in this context, but I prefer to at least order slice colors darkest to lightest for biggest to smallest, or vice versa), please see Reference 4. If an ODS GRAPHICS method is ever provided to create pie charts, the ODS GRAPHICS Attribute Mapping feature could be used to control slice color, but, even then, some preprocessing of the data would be required, and the legend entries would require a ranking numeric prefix.

## TIME PLOTS, TIME SERIES GRAPHS, TREND LINES, LINE CHARTS, etc.

In SAS/GRAPH, the GPLOT procedure can visually present the evolution of data values over time. In ODS GRAPHICS, the SG procedures provide more ways. In References 5 & 6, in collaboration with Alexandra Riley, I compared ways to visually present time series data using the old and new technology. These examples use web-deployed graphs so that ALT text can be provided, rather than forcing the viewer to guess y and x values based on tick mark values at the axes. Another solution for time series, available in Reference 3, is to link the plot forwards and backwards with a spreadsheet. Image height and width for SG procedure output are the ODS GRAPHICS defaults.

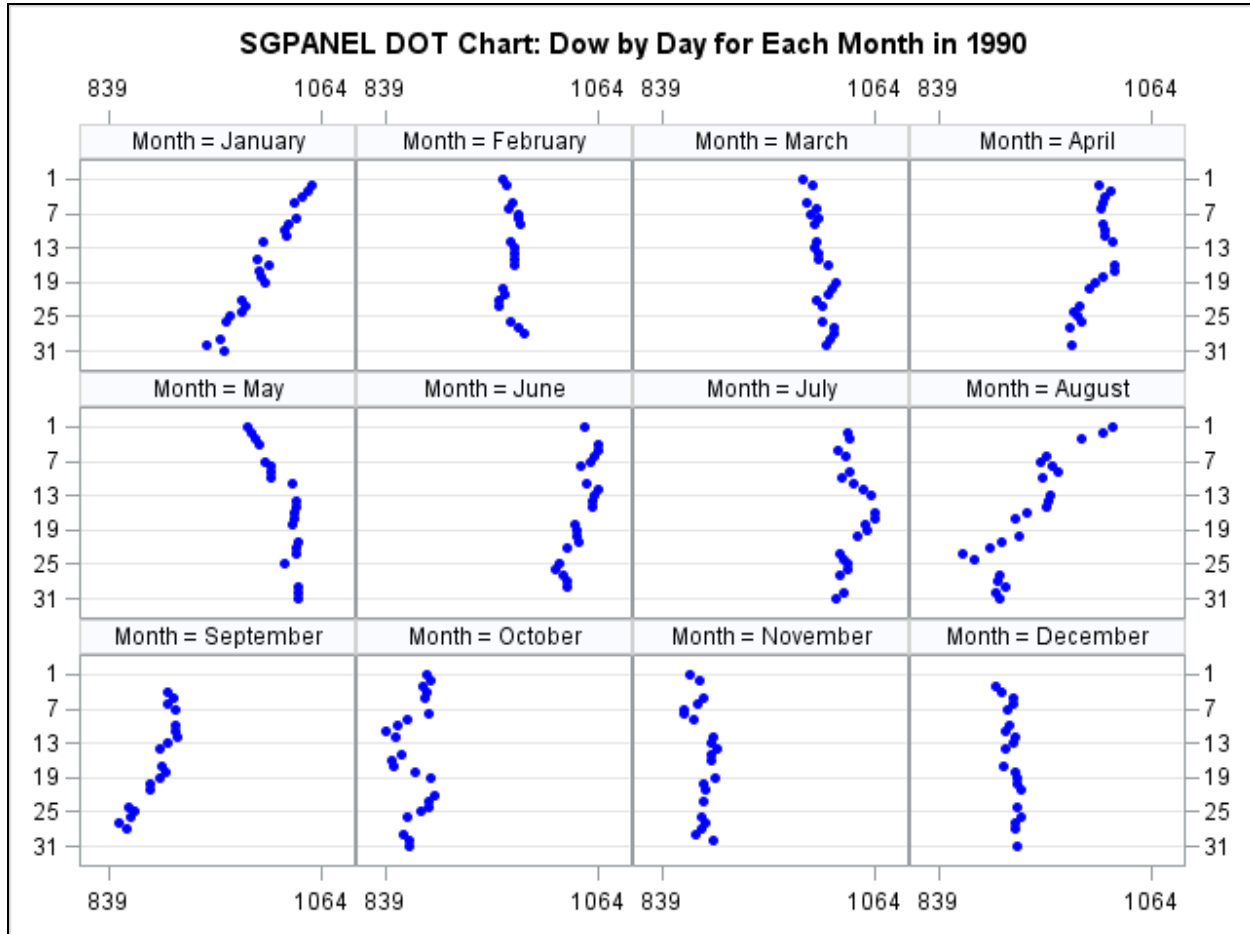
**NOTE 1:** Except for HTMLPANEL example, the illustrations in this section are inserted PNG files, not screen prints of the web graphs from a web browser window. However, all code does create the web-enabling HTML for the graphs.  
**NOTE 2:** These examples require that the code in Appendix 1 be run first.

In these examples, DESCRIPTION=' ' is used in order to prevent the display of the default pop-up text that tries to describe the graph. I chose not to provide a customized description.

The ALT text for PROC GPLOT is provided via the PLOT statement's HTML parameter, which identifies a variable on the plot input data set. The variable can be customized with anything the designer/programmer desires, including line breaks, any text, and the plot point values in any format to any precision, e.g. with day of the week name, as well as

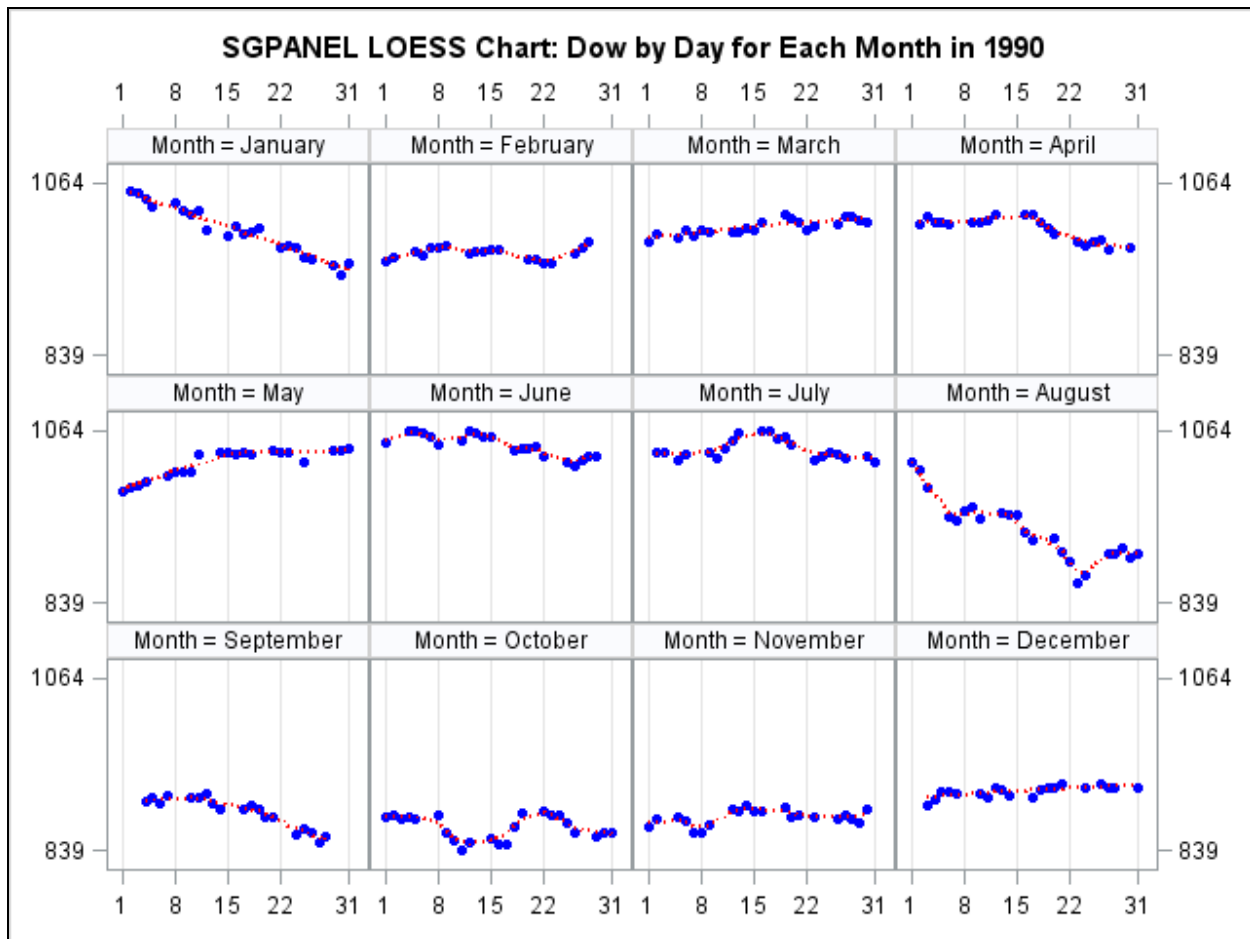
month, day, year. For Statistical Graphics procedures, ALT text is triggered by IMAGEMAP=ON in an ODS GRAPHICS statement. The text displayed is simply a list of the form Variable Name (or Label) = Value, and cannot be customized.

Since the best way to determine the precise value of the Dow in these plots is from pop-up ALT text, in most cases here I display only the year's minimum and maximum values for the Dow. PROC SGSCATTER allows no control.



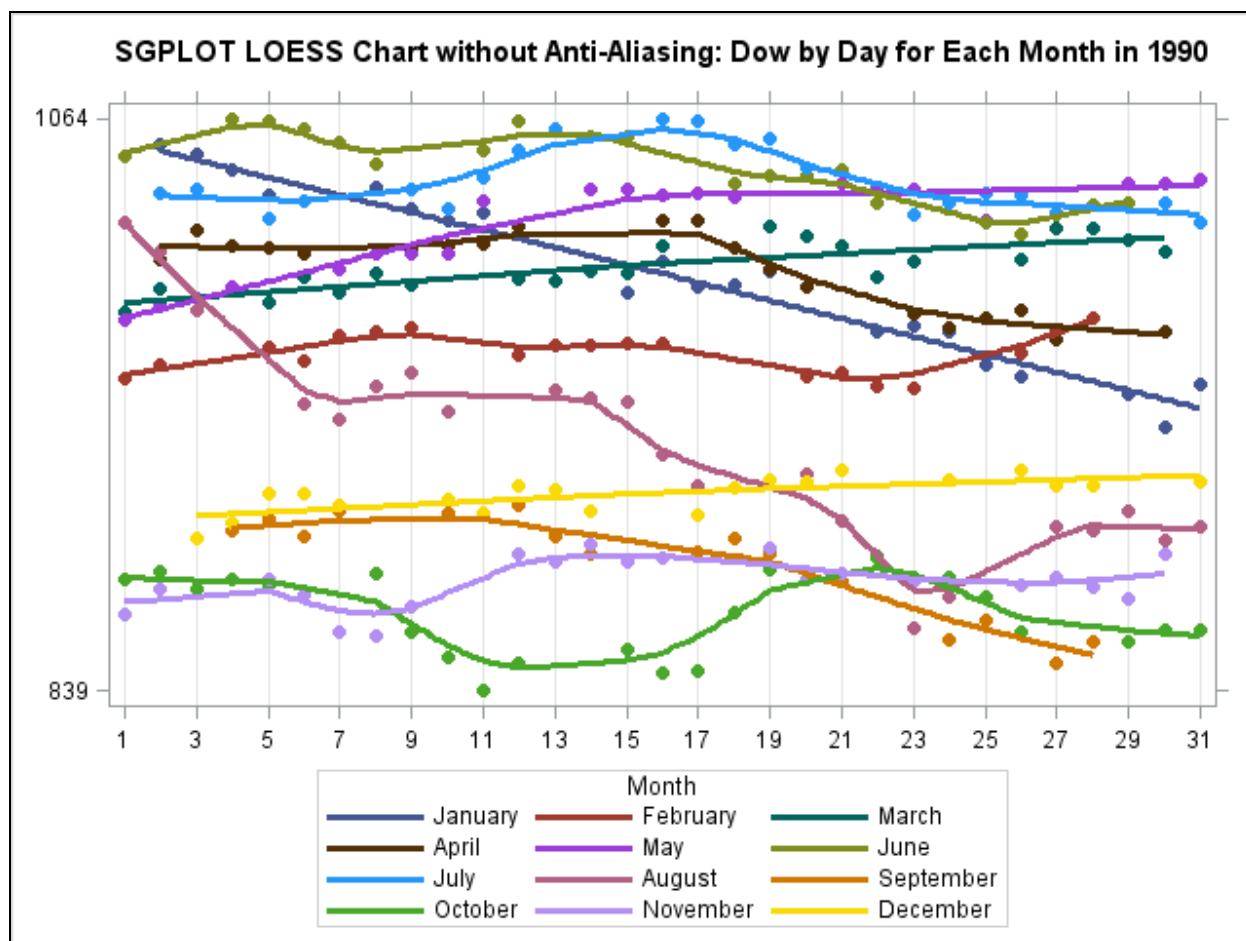
```
ods noresults; ods listing close;
ods graphics on / reset=all border=on imagemap=on imagename='SGPANEL_DOT';
ods html path="C:\Graph Alternatives Old and New" (url=none)
  body='SGPANEL_DOT.html' (title='SGPANEL DOT Chart Dow by Day Each Month in 1990');
title 'SGPANEL DOT Chart: Dow by Day for Each Month in 1990';
proc sgpanel data=work.DowByDayIn1990;
  panelby Month / columns=4 rows=3;
  dot Day / response=Dow nostatlabel
    markerattrs=(color=blue size=5 symbol=circlefilled);
  rowaxis display=(nolabel) refticks=(values) values=(1 8 15 22 31) grid;
  colaxis display=(nolabel) refticks=(values) values=(839 1064 839 1064 839 1064);
format Month MonthNm.;
format Dow 5. Day 2.;
run; quit;
ods html close; ods listing;
```

For the day of the month, on displays that are too small to show all of the day numbers, I favor using tick mark values 1, 8, 15, 22, 31. For this SGPANEL DOT chart, the PROC disregarded my coding choice, without any message in the SAS log, and instead used equally spaced values 1, 7, 13, 19, 25, 31.



```
ods noresults;
ods listing close;
ods graphics on / reset=all border=on imagemap=on imagename='SGPANEL_LOESS'
                  antialiasmax=2500 tipmax=2500;
ods html path="C:\Graph Alternatives Old and New" (url=none)
        body='SGPANEL_LOESS.html'
        (title='SGPANEL LOESS Chart Dow by Day Each Month in 1990');
title 'SGPANEL LOESS Chart: Dow by Day for Each Month in 1990';
proc sgpanel data=work.DowByDayIn1990;
  panelby Month / columns=4 rows=3;
  loess y=Dow x=Day / nolegfit markerattrs=(color=blue size=8 symbol=circlefilled)
        lineattrs=(color=red pattern=dot);
  rowaxis display=(nolabel) refticks=(values) values=(&Ymin &Ymax);
  colaxis display=(nolabel) refticks=(values) values=(1 8 15 22 31) grid;
format Month MonthNm.;
format Dow 5. Day 2.;
run; quit;
ods html close;
ods listing;
```

For the day of the month, on displays that are too small to show all of the day numbers, I favor using tick mark values 1, 8, 15, 22, 31.



```
ods noresults; ods listing close;
ods graphics on / reset=all border=on antialias=off tipmax=2500
                  imagemap=on imagename='SGPLOT_LOESS_withoutAntiAliasing';
ods html path="C:\Graph Alternatives Old and New" (url=none)
      body='SGPLOT_LOESS_withoutAntiAliasing.html'
      (title='SGPLOT LOESS Chart without Anti-Aliasing Dow by Day Each Month in 1990');
title 'SGPLOT LOESS Chart without Anti-Aliasing: Dow by Day for Each Month in 1990';
proc sgplot data=work.DowByDayIn1990;
  loess y=Dow x=Day / group=Month markerattrs=(size=7 symbol=circlefilled)
        lineattrs=(thickness=3 pattern=solid);
  yaxis display=(nolabel) values=(&Ymin &Ymax);
  xaxis display=(nolabel) values=(1 to 31 by 1) grid;
format Month MonthNm.;
format Dow 5. Day 2.;
run; quit;
ods html close; ods listing;
```

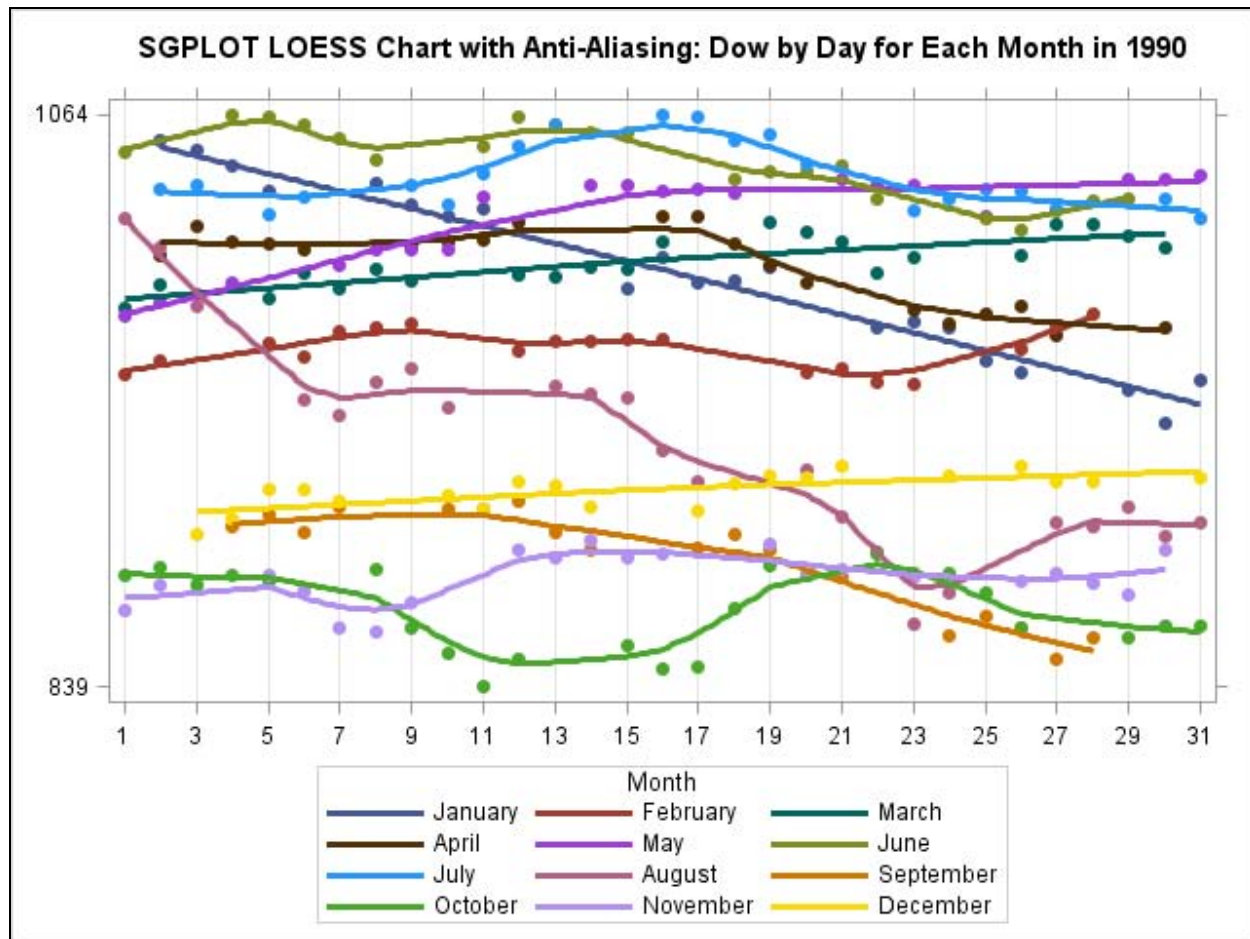
**NOTE 1:** refticks=(values) is not an available option for PROC SGPLOT and PROC SGSCATTER, and refticks without tick mark values serve no visual communication purpose if you have specified a grid.

**NOTE 2:** Anti-aliasing is a method to smooth lines that are not horizontal, and to smooth curves. It is more effective when the line or curve is thin. You will not see a dramatic improvement in the image on the next page. In V9.3, anti-aliasing is the default for the PNG driver. The PNG driver is the default for the ODS HTML destination.

**NOTE 3:** Color lines (and plot markers and legend samples) must be sufficiently thick (big) if you want viewers of your graph to be able to distinguish the colors.

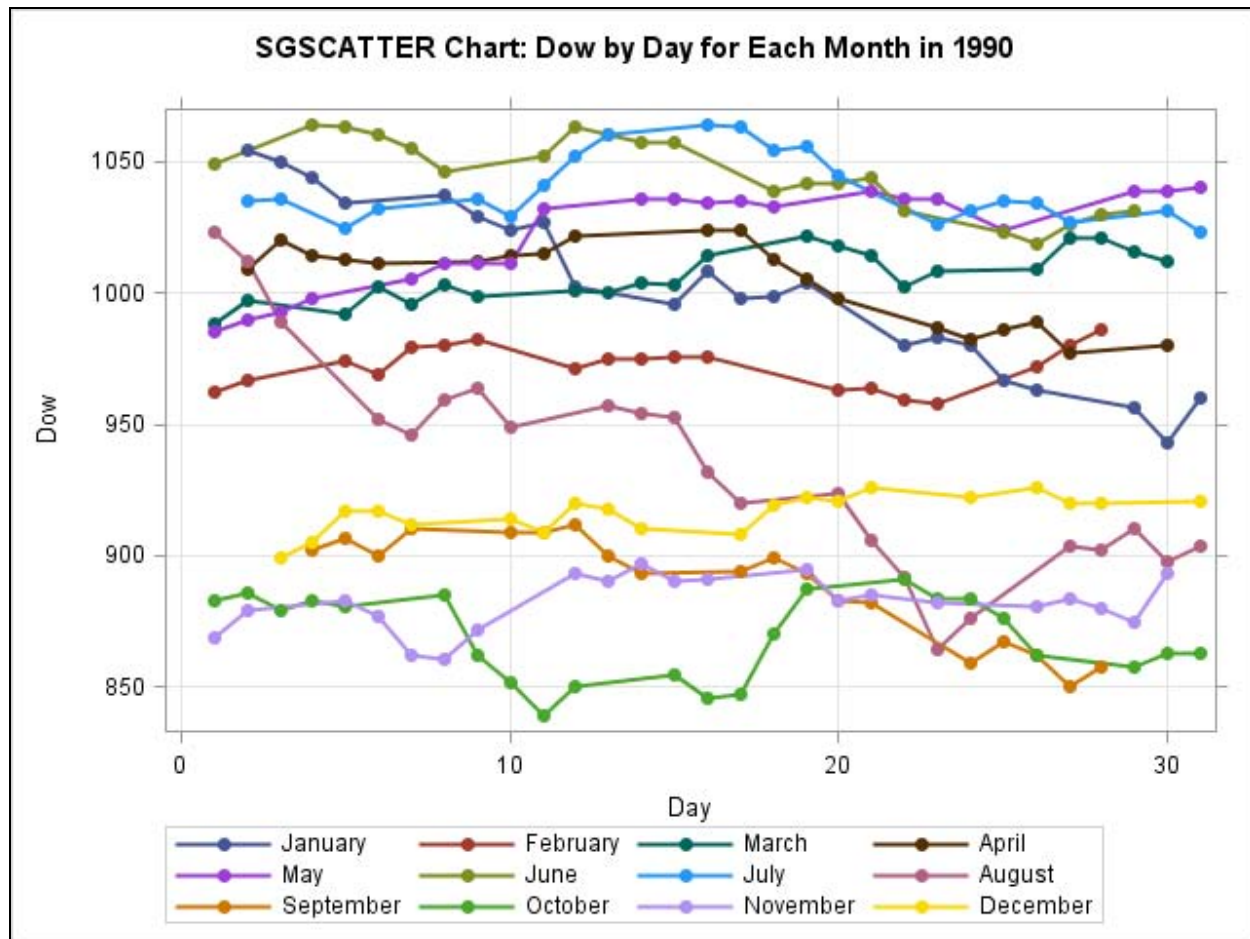
**NOTE 4:** Although there is enough space, this PROC disregarded my coding choice of x axis tick mark values 1, 2, 3, . . . , 31, without any message in the SAS log, and used only every other day instead.





```
ods noresults; ods listing close;
ods graphics on / reset=all border=on antialiasmax=2500 tipmax=2500
                  imagemap=on imagename='SGPLOT_LOESS_withAntiAliasing';
ods html path="C:\Graph Alternatives Old and New" (url=none)
      body='SGPLOT_LOESS_withAntiAliasing.html'
      (title='SGPLOT LOESS Chart with Anti-Aliasing Dow by Day Each Month in 1990');
title 'SGPLOT LOESS Chart with Anti-Aliasing: Dow by Day for Each Month in 1990';
proc sgplot data=work.DowByDayIn1990;
  loess y=Dow x=Day / group=Month
        markerattrs=(size=7 symbol=circlefilled)
        lineattrs=(thickness=3 pattern=solid);
  yaxis display=(nolabel) values=(&Ymin &Ymax);
  xaxis display=(nolabel) values=(1 to 31 by 1) grid;
format Month MonthNm.;
format Dow 5. Day 2.;
run; quit;
ods html close; ods listing;
```

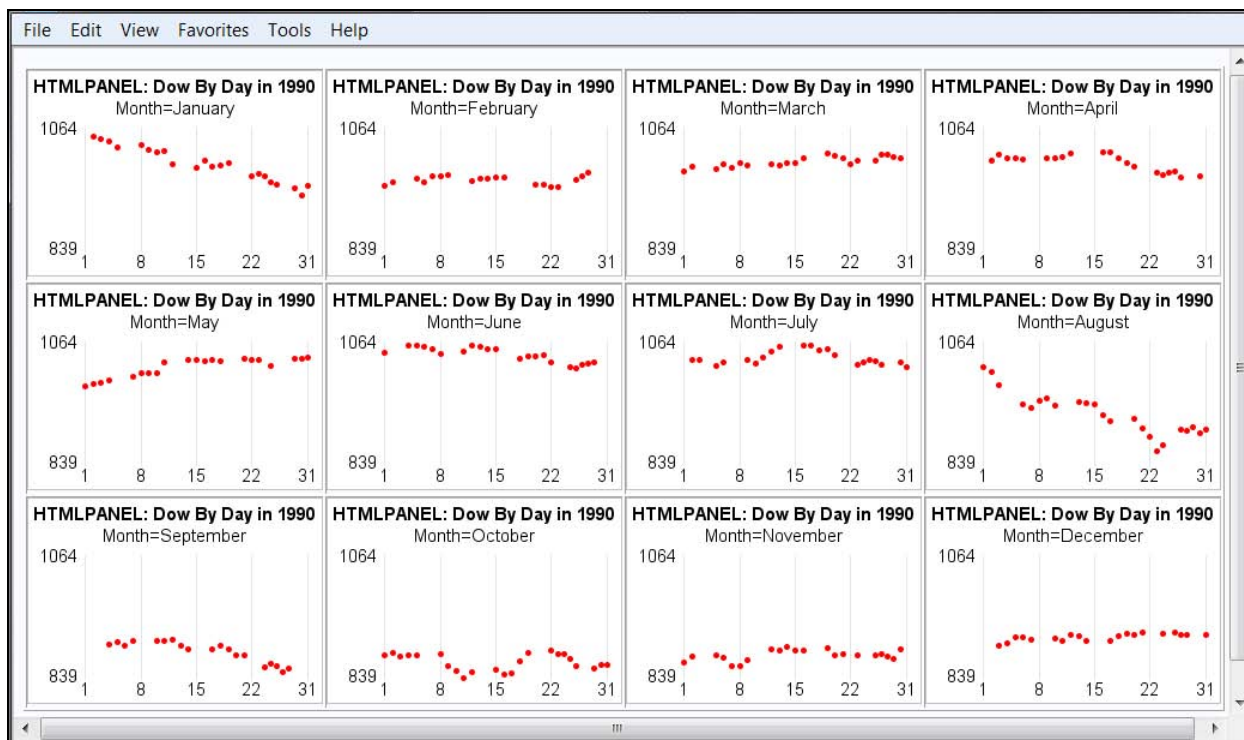
Turning on anti-aliasing, to get the slightly smoother curves, for this SGPLOT LOESS chart made it necessary to add **ANTIALIASMAX=2500** to the ODS GRAPHICS statement.



```
ods noresults;
ods listing close;
ods graphics on / reset=all border=on imagemap=on imagename='SGSCATTER';
ods html path="C:\Graph Alternatives Old and New" (url=none)
  body='SGSCATTER.html' (title='SGSCATTER Chart Dow by Day Each Month in 1990');
title 'SGSCATTER Chart: Dow by Day for Each Month in 1990';
proc sgscatter data=work.DowByDayIn1990;
plot Dow*Day / group=Month legend=(notitle) grid
  join=(lineattrs=(thickness=2 pattern=solid))
  markerattrs=(size=7 symbol=circlefilled);
format Month MonthNm.;
format Dow 5. Day 2.;
run; quit;
ods html close;
ods listing;
```

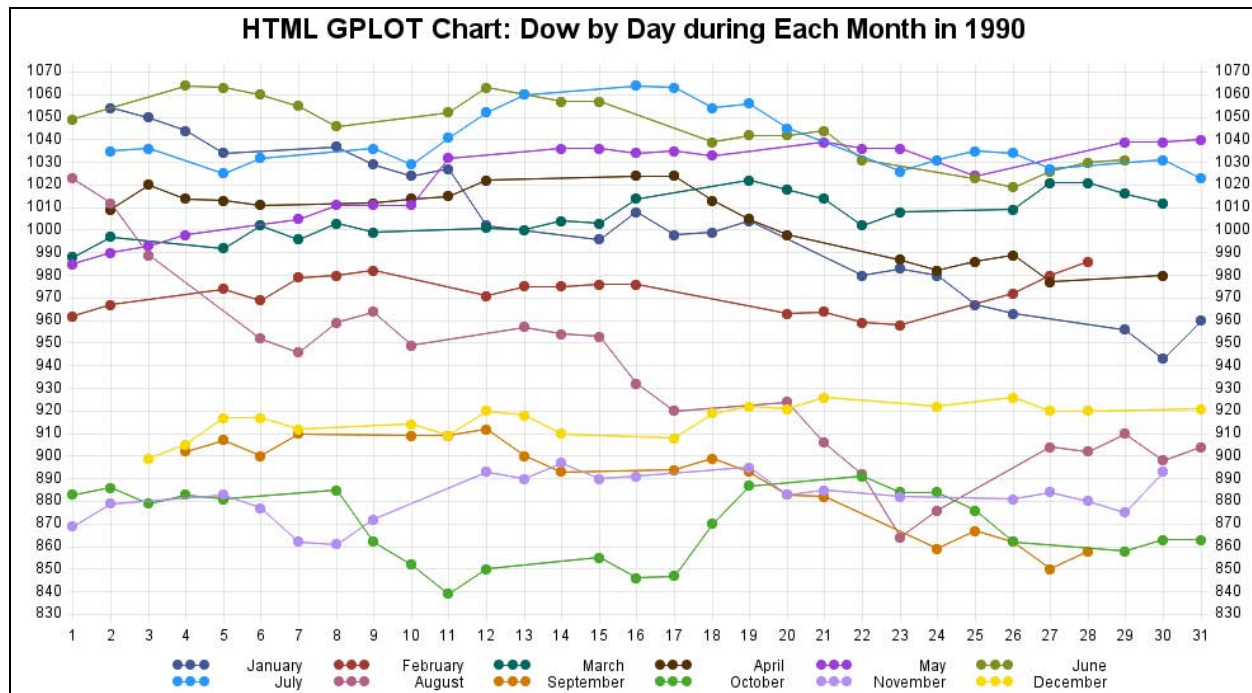
**NOTE 1:** PROC SGSCATTER provides NO programmer control of tick mark values.

**NOTE 2:** There is a DATALABEL option available on the PLOT statement for PROC SGSCATTER, which provides, by default, hard annotation of the y variable. However, on a multi-line plot it is impractical to use it due to overlaps with other lines and their annotation. However, even for a single-line plot, it might not do as good a job as my own alternative to the POINTLABEL option provided by the SAS developers for PROC GPLOT.



```
ods noresults;
ods listing close;
goptions reset=all;
ods tagsets.htmlpanel options(PanelColumns='4' PanelBorder='1')
  path="C:\Graph Alternatives Old and New" (url=none)
  style=HTMLblue /* the SAS developers like this style */
  body='HTMLPANEL_GPLOT.html'
  (title='HTMLPANEL GPLOT Chart Dow by Day Each Month in 1990');
title1 height=3 PCT ' ' justify=center height=9 PCT "HTMLPANEL: Dow By Day in 1990";
goptions htext=9 PCT;
goptions xpixels=245 ypixels=170 border;
ods tagsets.htmlpanel event=panel(start);
proc gplot data=work.DowByDayIn1990_withAltText;
  by Month;
  symbol1 interpol=none font='Webdings' value='6E'X height=1.5 color=red;
  axis1 label=none style=0 major=none minor=none order=&Ymin to &Ymax by &Yrange;
  axis2 label=none style=0 major=none minor=none order=1 8 15 22 31
    offset=(2 PCT, 2 PCT);
  plot Dow*Day / description=' ' html=AltTextVar
    vaxis=axis1 haxis=axis2 noframe autohref;
format Month MonthNm.;
format Dow 5. Day 2.;
run; quit;
ods tagsets.htmlpanel event=panel(finish);
ods tagsets.htmlpanel close;
ods listing;
```

"Dow by Day in 1990" is repeated in EVERY panel, rather than being a title above the array, because the option EMBEDDED\_TITLES='NO' does not work as documented. The BYLINES='NO' option does not work either, but, in this case, the bylines are desired. If the HTMLPANEL tagset worked as documented, it would provide a much simpler way to emulate what can be done so well, but with much more code, using PROC GREPLAY. To list in the SAS log all HTMLPANEL options theoretically available, include doc='help' in the ods tagsets.htmlpanel options. There is no information about HTMLPANEL in the SAS OnlineDoc. Instead, see <http://support.sas.com/kb/38/066.html> and <http://support.sas.com/rnd/base/ods/odsmarkup/htmlpanel.html>.



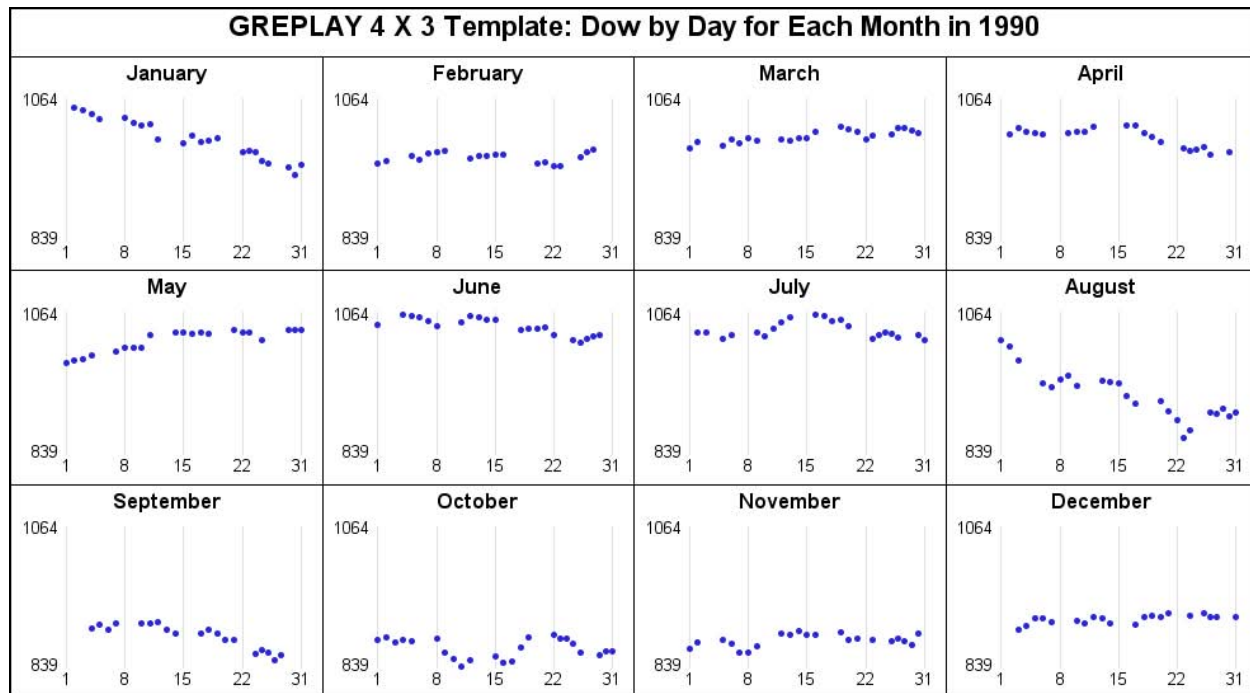
```
ods noresults;
ods listing close;
goptions reset=all;
ods html path="C:\Graph Alternatives Old and New" (url=none)
    body='HTML_GPLOT.html' (title='HTML GPLOT Chart Dow by Day Each Month in 1990');
goptions htext=2.25 PCT xpixels=983 ypixels=544 border;
title1 height=1 PCT ' ' justify=center height=4 PCT
    'HTML GPLOT Chart: Dow by Day during Each Month in 1990';
proc gplot data=work.DowByDayIn1990_withAltText;
    symbol1 interpol=join font='Webdings' value='6E'X height=0.75 r=12;
    axis1 label=none style=0 major=none minor=none
        order=&RoundedYmin to &RoundedYmax by &Yincrement;
    axis2 label=none style=0 major=none minor=none order=1 to 31 by 1;
    legend1 label=none shape=symbol(3,0.75);
    plot Dow*Day=Month / name='OneGPLOT' description=' ' html=AltTextVar
        vaxis=axis1 haxis=axis2 noframe autovref autohref legend=legend1;
    plot2 Dow*Day=Month / vaxis=axis1 nolegend;
format Month MonthNm.;
format Dow 5. Day 2.;
run; quit;
ods html close; ods listing;
```

**NOTE 1:** Only this GPLOT Overlay Chart is willing to display a tick mark for every day of the month. In the SGPLOT LOESS Chart examples, the software disregarded my coded instructions without any message in the SAS log.

**NOTE 2:** In this plot, contrary to my usual preference, I chose to display more vertical tick mark values and to provide reference lines at every one of those values.

**NOTE 3:** The image sizing xpixels=983 ypixels=544 for this graph and for that on the next page created with PROC GREPLAY was optimized to let these graphs fill the available space in an Internet Explorer web browser window on a monitor with resolution 1024 X 768. Web graphs are best sized when they are as large as possible, but require no scrolling to view them in full.





For the lengthy code to create the above, please see Appendix 2.

## CONCLUSION

ODS GRAPHICS and the SG procedures are very young technology, and users will need to experience numerous versions before all of their obvious needs are addressed. It would have been useful if more capabilities were included in the graph types delivered, and if all obvious graph types were available (e.g., pie charts), rather than less urgently needed graph types and a capability deficit. Users would prefer all of their needs to be met with Version 1.

## REFERENCES

1. Bessler, LeRoy. "Multi-Media Wizardry: How To Make ODS Outputs That 'Dance and Sing'", *Proceedings of the Thirty-First Annual SAS Users Group International Conference*. Cary, NC: SAS Institute Inc. 2006. [www2.sas.com/proceedings/sugi31/090-31.pdf](http://www2.sas.com/proceedings/sugi31/090-31.pdf)
2. Summers, E., Allison, R., Langston, J., and Cowley, J. "Using SAS/GRAPH to Create Visualizations That Also Support Tactile and Auditory Interaction", *Proceedings of the SAS Global Forum 2012 Conference*. Cary, NC: SAS Institute Inc. 2012. <http://support.sas.com/resources/papers/proceedings10/231-2010.pdf>
3. Bessler, LeRoy. "The Most Communication-Effective and Most Usable Information Delivery", *Proceedings of the SAS Global Forum 2010 Conference*. Cary, NC: SAS Institute Inc. 2010. <http://support.sas.com/resources/papers/proceedings10/231-2010.pdf>
4. Bessler, LeRoy. "Communication-Effective Pie Charts", *Proceedings of the SAS Global Forum 2007 Conference*. Cary, NC: SAS Institute Inc. 2007. <http://www2.sas.com/proceedings/forum2007/134-2007.pdf>
5. Bessler, LeRoy. "All the Ways to Display Multiple Trend Charts", *IEWS News, Issue 52*. UK: IEVS International SAS Programmer Community. 2011. [http://www.sascommunity.org/wiki/IEWS\\_News](http://www.sascommunity.org/wiki/IEWS_News)
6. Bessler, LeRoy and Riley, Alexandra. "All the Ways to Display Multiple Trend Charts: Three Selected Examples and Their Code", *IEWS News, Issue 53*. UK: IEVS International SAS Programmer Community. 2011. [http://www.sascommunity.org/wiki/IEWS\\_News](http://www.sascommunity.org/wiki/IEWS_News)

## CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

LeRoy Bessler PhD  
 Bessler Consulting and Research  
 262-242-1099  
[Le\\_Roy\\_Bessler@wi.rr.com](mailto:Le_Roy_Bessler@wi.rr.com)

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. © indicates USA registration.  
Other brand and product names are trademarks of their respective companies.

## APPENDIX 1. Set-Up Code for Time Series Graphs

**NOTE:** The citiday data set is automatically present in the SASHELP data library only if SAS/ETS is licensed. However, you might be able to get five data sets citiday, citiweek, citimon, citiqtr, and citiyr from SAS Technical Support.

### Used by All Time Series Graphs Examples

```
proc format library=work;
  value MonthNm
    1 = 'January'
    2 = 'February'
    3 = 'March'
    4 = 'April'
    5 = 'May'
    6 = 'June'
    7 = 'July'
    8 = 'August'
    9 = 'September'
   10 = 'October'
   11 = 'November'
   12 = 'December';
run; quit;

data work.DowByDayIn1990(keep=Year Month MonthAbbrev MonthName Day Dow date);
format Dow 5.;
set sashelp.citiday(keep=date snydjcm where=(snydjcm ne .));
Year = year(date);
if Year EQ 1990;
Month = month(date);
MonthAbbrev=put(date,monname3.);
MonthName=put(date,monname9.);
Day = day(date);
Dow=round(snydjcm,1);
run;
```

### Used by HTMLPANEL, GPLOT, and GREPLAY Examples

```
data work.DowByDayIn1990_withAltText;
length AltTextVar $ 200;
set work.DowByDayIn1990;
AltTextVar = 'alt="On ' || trim(left(put(date, WEEKDATX.))) ||
'0D'X || 'the Dow was ' || trim(left(put(Dow, 5.))) || '"';
/* '0D'X forces a line break */
run;
```

### Used by SGPANEL and SGPLOT Examples

```
proc means data=work.DowByDayIn1990 min max range noprint;
var Dow;
output out=minmax min=DowMin max=DowMax range=DowRange;
run;

data _null_;
set minmax;
call symput('Ymin' ,trim(left(put(DowMin ,5.))));
call symput('Ymax' ,trim(left(put(DowMax ,5.))));
call symput('Yrange',trim(left(put(DowRange,5.))));
run;
```



Used Only by GREPLAY Example

```

data _null_;
/* This logic is suitable for the KNOWN and RATHER LIMITED range of this data set:
   bounded by 830 and 1070. It is intended as a POSSIBLE model for enhanced reuse
   in generalized processing. Too many tick marks can cause graph infeasibility. */
retain ActualYmax &Ymax ActualYmin &Ymin;
set minmax;
call symput('Yincrement',10);
TentativeRoundedYmax = round(ActualYmax,10);
if ActualYmax LE TentativeRoundedYmax
then RoundedYmax = TentativeRoundedYmax;
else RoundedYmax = TentativeRoundedYmax + 10;
call symput('RoundedYmax',trim(left(put(RoundedYmax,5)))));
TentativeRoundedYmin = round(ActualYmin,10);
if ActualYmin GE TentativeRoundedYmin
then RoundedYmin = TentativeRoundedYmin;
else RoundedYmin = TentativeRoundedYmin - 10;
call symput('RoundedYmin',trim(left(put(RoundedYmin,5)))));
run;

```

**APPENDIX 2. Using PROC GREPLAY to create an array of multiple GPLOT outputs**

```

goptions reset=all;
goptions device=png;
* goptions cback=green; /* This WOULD be the background color for the template PREVIEW
in PROC GREPLAY below if UNcommented here, and then WOULD be the background color for
the graph when the template is used in GREPLAY, unless overridden at that time.
It could be overridden then with GOPTIONS RESET or with GOPTIONS CBACK=OtherColor. */
goptions xpixels=983 ypixels=544; /* defaults for GIF are 800 and 600 */

proc greplay tc=work.tempcat nofs; /* tempcat = "template catalog",
    not "temporary catalog" but, being in the WORK folder, it IS temporary. */

    tdef Wide4Hi3 des='Four Wide Three High'

        1/llx=000 lly=0   ulx=000 uly=31 urx=025 ury=31 lrx=025 lry=0
        2/llx=025 lly=0   ulx=025 uly=31 urx=050 ury=31 lrx=050 lry=0
        3/llx=050 lly=0   ulx=050 uly=31 urx=075 ury=31 lrx=075 lry=0
        4/llx=075 lly=0   ulx=075 uly=31 urx=100 ury=31 lrx=100 lry=0
        5/llx=000 lly=31  ulx=000 uly=62 urx=025 ury=62 lrx=025 lry=31
        6/llx=025 lly=31  ulx=025 uly=62 urx=050 ury=62 lrx=050 lry=31
        7/llx=050 lly=31  ulx=050 uly=62 urx=075 ury=62 lrx=075 lry=31
        8/llx=075 lly=31  ulx=075 uly=62 urx=100 ury=62 lrx=100 lry=31
        9/llx=000 lly=62  ulx=000 uly=93 urx=025 ury=93 lrx=025 lry=62
        10/llx=025 lly=62 ulx=025 uly=93 urx=050 ury=93 lrx=050 lry=62
        11/llx=050 lly=62 ulx=050 uly=93 urx=075 ury=93 lrx=075 lry=62
        12/llx=075 lly=62 ulx=075 uly=93 urx=100 ury=93 lrx=100 lry=62
        /* Container for Titles, Footnotes, and Panels 1-12 */
        13/llx=0   lly=0   ulx=0 uly=100 urx=100 ury=100 lrx=100 lry=0;

    template Wide4Hi3;
    list template;
run; quit;

goptions reset=all;
goptions device=png;
goptions NODisplay; /* Want to suppress unneeded graphics stream file while writing to
graphics catalog. Even with the fileref for gsfname cleared, a default gsf would be
written to My SAS Files */

goptions xpixels=245 ypixels=169;
goptions xpixels=980 ypixels=507;

```

```

goptions htext=8 PCT border;

data _null_;
do i = 1 to 12 by 1;
  call symput('MonthName' || trim(left(i)), trim(left(put(i, MonthNm.))));
end;
run;

%macro MakeMonthlyGraphs;

%do i = 1 %to 12 %by 1;
title1 height=2 PCT ' ' justify=center height=10 PCT "&&MonthName&i";
footnote1 height=0.5 PCT ' ';
footnote2 angle=-90 height=1 ' ';
footnote3 angle=+90 height=1 ' ';
proc gplot gout=work.gcat data=work.DowByDayIn1990_withAltText;
  where month(date) EQ &i ;
  symbol1 interpol=none font='Webdings' value='6E'X height=1.5;
  axis1 label=none style=0 major=none minor=none order=&Ymin to &Ymax by &Yrange;
  axis2 label=none style=0 major=none minor=none order=1 8 15 22 31
    offset=(2 PCT, 2 PCT);
  plot Dow*Day / name="Month&i" description=' ' html=AltTextVar
    vaxis=axis1 haxis=axis2 noframe autohref;
run; quit;
%end;

%mend MakeMonthlyGraphs;

%MakeMonthlyGraphs;

goptions xpixels=983 ypixels=544 border;
proc gslide gout=work.gcat name="TtlFtnt" description="Titles and Footnotes";
title1 height=0.5 PCT ' ' justify=center height=4 PCT
  "GREPLAY 4 X 3 Template: Dow by Day for Each Month in 1990";
run; quit;

goptions reset=all;
* goptions cback=blue; /* This WOULD be the background color for the graph
                        if UNcommented here. */

ods noresults;
ods listing close;
ods html path="C:\Graph Alternatives Old and New" (url=none)
  body='GREPLAY_4X3_GPLOT.html'
  (title='GREPLAY 4 X 3 Template Chart Dow by Day Each Month in 1990');
goptions xpixels=983 ypixels=544 border;
title1; footnote1;
goptions display;
proc greplay igout=work.gcat nofs;
  tc work.tempcat;
  template Wide4Hi3;
  treplay
    9:Month1 10:Month2 11:Month3 12:Month4
    5:Month5 6:Month6 7:Month7 8:Month8
    1:Month9 2:Month10 3:Month11 4:Month12
    13:TtlFtnt;
run; quit;
ods html close; ods listing;

```