

Data Visualization Power Tools: Expedite the Easy, Implement the Difficult, or Handle Big Data

LeRoy Bessler

Bessler Consulting and Research, Mequon, Milwaukee, Wisconsin, USA

ABSTRACT

The power tools are SAS macros for ODS Graphics or SAS/GRAPH. You can use them As Is, modify them for similar, but different, tasks, or learn from them to build new solutions for very different problems. Power tools enable you to create results quickly and easily. Power tools that are poorly designed enable you to create scrap quickly and easily. Here, the standard of good design is communication effectiveness of the result. Some of the tools presented can enable you to show what's important when confronted with too much information, and to fit whatever is of importance into the web browser window, preventing any need for scrolling. To interpret a picture, you need to see the whole picture at one time. Other tools presented, which are not web-specific, but can be used in that context, enable you to get the best result out of SAS graphic software with minimal coding.

INTRODUCTION

For over thirty years starting with SAS/GRAPH and recently also with ODS Graphics and the new SG (Statistical Graphics) Procedures, I have been on a quest in pursuit of data insights and visual communication effectiveness.

These macro power tools help you get the best out of SAS for data visualization.

For my first several years as a SAS programmer, I thought that SAS macro language looked peculiar, and wondered why anyone would insist on using it. Then I finally encountered a data visualization situation where SAS Technical Support explained that use of macro language and macro variables was the ONLY way to achieve my design and operational objective. I solved that problem, but shortly thereafter discovered that SAS macro language could also be used in what I called Software-Intelligent Application Development. Though that topic is outside the scope of this paper, let me emphasize that SIAD can maximize Reliability, Reusability, Maintainability, Extendibility, and Flexibility. I have written about this concept several times over the years, between 1992 and 2008. See References 1 and 2 for the earliest and latest papers. In Reference 2, you can find my two theorems, my corollary, and my formula that capture the reality of program maintainability as my contribution to "Programming Science". They were first published in 1998 in a paper called "Software Intelligence: Applications That Customize Themselves".

Macros can get you beyond the defaults, and deliver the right result the first time every time, with no iterative hands-in intervention. They are essential not only for production batch processing, but also for real-time response to queries.

My tools are for 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 series plots (or trend charts), bar charts, and pie charts.

Some of the macro tools make it easier and safer to create the results. Other of the macro tools make widely applicable, high-value solutions possible. Some of the tools are expressly designed for the new ODS Graphics / SG (Statistical Graphics) Procedures. Three related tools exploit the capability of the new technology to deliver the results in a nicer package. Other of the tools can be implemented using either the new technology or SAS/GRAPH, but, until the new technology is improved, the SAS/GRAPH version of the macro is the wiser choice.

In a sense, some parts of this paper are a workshop or laboratory where I step through the process of getting to a best result, and take the reader along with me. An easy way to get through this long paper is to focus on the illustrations and the macro invocations, and ignore the details inside the macro code unless you want to change it. Take what you like, and leave the rest.

My thanks to Laura MacBride for her helpful review of this paper.

Related and available in the PharmaSUG 2013 Conference Proceedings is a companion paper, which is macro-free, on guidelines and methods for communication-effective data visualization (Reference 3).

GRAPH MACRO TOOLS YOU CAN USE IN SAS V9.3

My quest since 1981 for communication-effective graphs has yielded non-default solutions that require extra coding. Macros can do that coding work for you, and give you simple parameters to specify instead, similar to using a SAS procedure. In this section, I share and show how to use such macros for common uses with high-value results.

Two of the macros, one for a pie chart and the other for a bar chart to serve as a pie chart alternative when a pie chart is infeasible or not desired, overcome limitations in the new ODS GRAPHICS and SG procedures which are now an alternative to SAS/GRAPH. With those macros you need not use SAS/GRAPH to compensate for what is missing in the new technology. Their output charts are written to disk to later be inserted into Microsoft PowerPoint, Word, or Excel, or simply printed, but they could be web-enabled and packaged with ODS HTML. In Reference 3, corresponding non-macro graphic solutions were provided to meet these exact same needs. Those solutions are made easier and less error vulnerable by use of macros provided below.

Big Data is not, as (I like to, with tongue in cheek, claim that) I first thought, character strings rendered all in upper case. Nor is it numeric data at the digit-string-length limit of what SAS software can safely handle. I assume that the Big Daters just wanted a shorter phrase or something cuter sounding than “Voluminous Data”.

NOTE: In this paper, **Big Data** means: (a) categorical data with so many distinct keys that a bar chart could be unwieldy—unless dealing with it as I recommend; or (b) time series data that contains so many date/datetime/time values that getting The Big Picture, yet being able to inspect the granularity, is a challenge.

The third and fifth macros in this paper, for a Maximally Informative Subsetted and Ranked Horizontal Bar Chart, do create web-enabled output. The fifth macro is a more complex, but more powerful and more flexible, version of the third macro, and it optionally provides hyperlinks to supporting data (from the bar chart to a spreadsheet of its input, and from any bar to a spreadsheet of the raw data that was summarized, or was used As Is, for that bar).

The fourth macro, also for a Maximally Informative Subsetted and Ranked Horizontal Bar Chart, is intended to let you custom format and size the output chart on disk for post-creation manual insert to Word or PowerPoint.

Remaining macros (provided both for the new technology, and for SAS/GRAPH—because SAS/GRAPH performs better, as explained there) enable you to handle ultra-wide trend charts in a communication-effective manner.

Some of the macro code is very lengthy and complex, but once you have copied it from here and filed it in a macro library, its use is easy and quick. Sample invocations of the macros are used here to create output examples.

You do not need to read any of macro source code provided here. If you do, obviously you can use it as a starting point to create a derivative or alternative that you like better.

How To Use the Macros

Copy the code for each macro into NotePad, and save the result with filetype sas in any folder location convenient to you where you have write access as well as read access. The filename that you use must match the string after the %MACRO prefix in the first line of the macro.

To run the macros, you must, as shown in the macro invocation example code, precede the macro invocation with this statement:

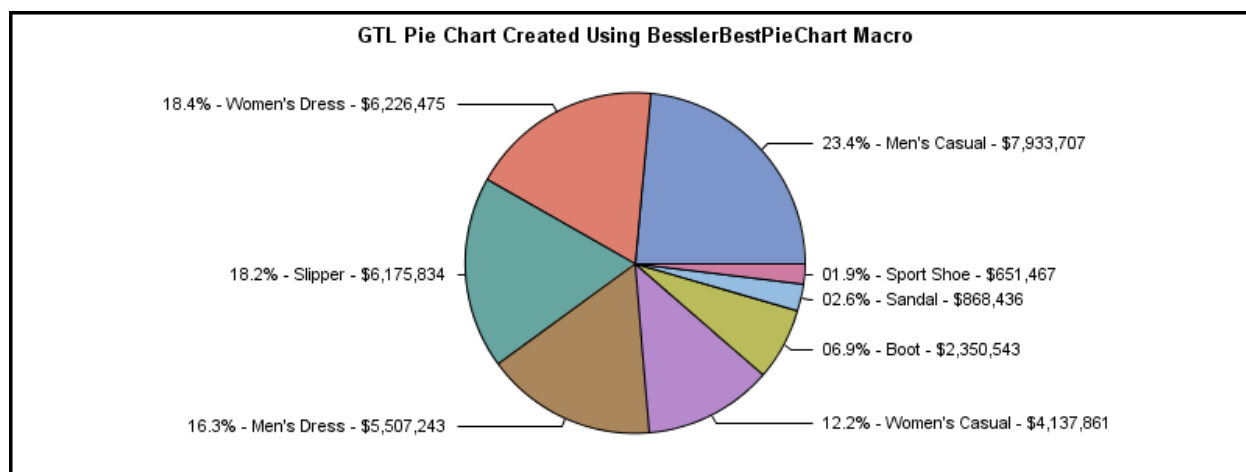
```
OPTIONS SASAUTOS=( ThePathToYourFolderGoesHere SASAUTOS );
```

An example of **ThePathToYourFolderGoesHere** could be:

```
"C:\MySASmacros"
```

When you submit code to run the macro, SAS will first look for it in your folder, and then it will look for it in your default macro library location, which has path reference SASAUTOS. For every macro that is invoked during your session or in your batch program execution, until you override the above OPTIONS statement, SAS will always look in your macro library first.

TOOL 1. PIE CHART WITH ORDERING AND ALL RELEVANT NUMERIC INFORMATION



There is NOTHING ELSE you could need visually or numerically to have complete understanding of the information at this level, and ordering makes it easy to quickly assess the relative size of the shares of the whole.

Within the ODS GRAPHICS environment of Base SAS, there is no SG procedure to create a pie chart. Instead, a pie chart requires use of the cumbersome, and otherwise unnecessary, Graph Template Language (GTL). Below is the code for invocation of a macro to simplify the task of creating the pie chart above:

```
options sasautos=("D:\MySASmacros" sasautos);
/* First look in D:\MySASmacros for any macro to be invoked.
   If Not Found, then look in the default sasautos,
   which is the macro library shipped by SAS Institute,
   and maybe additional local SAS macro library(ies) at your site
   which has (have) been added to your site's default sasautos.
   You can reverse the order of the search list above.
   You can include additional custom macro libraries in the search list above.
   If the same macro name is used in multiple libraries,
   the first one found is used. */
```

```
options mprint;
```

```
%BesslerBestPieChart27Jun2012
(Data=sashelp.shoes
,SliceLabelVar=Product
,SliceMeasureVar=Sales
,SliceMeasureFormat=dollar10.
,Order=Descending
,ChartTitle=GTL Pie Chart Created Using BesslerBestPieChart27Jun2012 Macro
,ChartFileName=PieChartCreatedUsingBesslerBestPieChart27Jun2012Macro
,ChartFolderName=D:\!PharmaSUG2013\Results
,ChartHeight=300px
,ChartWidth=800px);
```

Here is the macro that was stored in D:\MySASmacros with filename BesslerBestPieChart27Jun2012.sas .

```
%macro BesslerBestPieChart27Jun2012
(Data=
,SliceLabelVar=
,SliceMeasureVar=
,SliceMeasureFormat=
,DecimalPositionsForPercents=1 /* can be 0 or any integer */
,Order= /* valid values are descending or ascending */
,ChartTitle=
```

```

,ChartFileName=
,ChartFolderName=
,ChartHeight=
,ChartWidth=
);

* All parameters above, except Order are mandatory *;
* If Order is not specified, the SAS PROC SORT default is ascending *;

proc summary data=&Data nway;
class &SliceLabelVar;
var &SliceMeasureVar;
output out=ToPrep sum=TotalByClass;
run;

proc sql noprint;
select sum(TotalByClass) into :GrandTotal from ToPrep;
quit;

data ToChart;
length SliceNameWithPercentAndValue $ 256; /* over-sized, but that is harmless */
set ToPrep;
SliceNameWithPercentAndValue =
  trim(left(
    put(((TotalByClass / &GrandTotal) * 100),z4.&DecimalPositionsForPercents)
  )) ||
  '% - ' ||
  trim(left(&SliceLabelVar)) ||
  ' - ' ||
  trim(left(put(TotalByClass,&SliceMeasureFormat)));
run;

proc sort data=ToChart;
by &Order TotalByClass;
run;

proc template;
define statgraph BesslerBestPieChart27Jun2012;
  begingraph;
    entrytitle "&ChartTitle";
    layout region;
      piechart category=SliceNameWithPercentAndValue
               response=TotalByClass /
               datalabelcontent=(category)
               datalabellocation=callout
               otherslice=FALSE;
    endlayout;
  endgraph;
end;
run;

ods listing gpath="&ChartFolderName";
ods graphics on / reset=all
                 border=on
                 height=&ChartHeight
                 width=&ChartWidth
                 imagename="&ChartFileName";

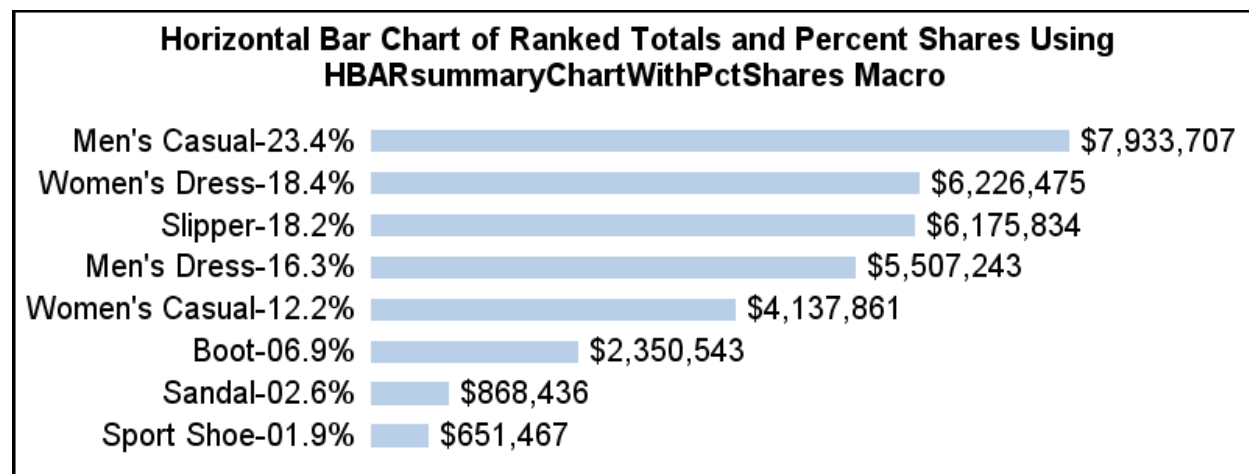
proc sgrender data=ToChart template=BesslerBestPieChart27Jun2012;
run;
ods listing close; ods listing;

%mend BesslerBestPieChart27Jun2012;

```

TOOL 2. BAR CHART OF SUMS WITH PERCENT SHARES: ALTERNATIVE FOR WHEN PIE CHART IS INFEASIBLE OR UNACCEPTABLE

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.



Just as the pie chart labels included values, not just slice names and percent shares, bar chart labels created in this macro COULD include values also, and the values at bar ends could be dropped. (The code for the Subsetted and Ranked Horizontal Bar Chart later in this paper will construct multi-function bar chart labels.)

The main advantage of SGPLOT bar charts is that you can get the values at the bar ends with no extraordinary effort. A PROC GCHART bar chart by default places those values in a stack or table at the right margin of the graph area. To get them at the bar ends requires use of the ANNOTATE facility, which is more work—best avoided if possible.

Below is the code for invocation of a macro to simplify the task of creating the bar chart above:

```
options sasautos=("D:\MySASmacros" sasautos);
/* First look in D:\MySASmacros for any macro to be invoked.
   If Not Found, then look in the default sasautos,
   which is the macro library shipped by SAS Institute,
   and maybe additional local SAS macro library(ies) at your site
   which has (have) been added to your site's default sasautos.
   You can reverse the order of the search list above.
   You can include additional custom macro libraries in the search list above.
   If the same macro name is used in multiple libraries,
   the first one found is used. */
options mprint;
%HBARsummaryChartWithPctShares
(Data=sashelp.shoes
,BarLabelVar=Product
,BarMeasureVar=Sales
,FontSize=16pt
,BarWidth=0.5
,Order=Descending
,ChartTitle=Horizontal Bar Chart of Ranked Totals and Percent Shares Using
HBARsummaryChartWithPctShares Macro
,ChartFileName=BarChartCreatedUsingHBARsummaryChartWithPctSharesMacro
,ChartFolderName=D:\PharmaSUG2013\Results
,ChartHeight=303px /* NOTE: At 300px, every other bar label is omitted by SAS. */
,ChartWidth=800px); /* See discussion of FitPolicy=THIN in Reference 3. */
```

Here is the macro that was stored in D:\MySASmacros with filename HBARsummaryChartWithPctShares.sas .

```
%macro HBARsummaryChartWithPctShares
(Data=
,BarLabelVar=
,BarMeasureVar=
,BarMeasureFormat=
,FontSize=
,BarWidth=
,DecimalPositionsForPercents=1 /* can be 0 or any integer */
,Order= /* valid values are descending or ascending */
,ChartTitle=
,ChartFileName=
,ChartFolderName=
,ChartHeight=
,ChartWidth=
);

%if %upcase(&Order) EQ DESCENDING
%then %let Order = respdesc;
%else %let Order = respasc ;

proc summary data=&Data nway;
class &BarLabelVar;
var &BarMeasureVar;
output out=ToPrep sum=TotalByClass;
run;

proc sql noprint;
select sum(TotalByClass) into :GrandTotal from ToPrep;
quit;

data ToChart;
length BarNameWithPercent $ 256; /* over-sized, but that is harmless */
set ToPrep;
BarNameWithPercent =
  trim(left(&BarLabelVar)) || '-' ||
  trim(left(put(((TotalByClass / &GrandTotal) * 100),z4.1))) || '%';
run;

proc template;
define style styles.ListingWithNoFrame; /* remove a useless box around the bars */
  parent=styles.Listing;
  class graphwalls / frameborder=off;
end; run;

ods listing gpath="&ChartFolderName" style=styles.ListingWithNoFrame;

ods graphics on / reset=all
  border=on height=&ChartHeight width=&ChartWidth imagename="&ChartFileName";

title height=&FontSize "&ChartTitle";
proc sgplot data=ToChart;
hbar BarNameWithPercent / response=TotalByClass categoryorder=&Order
  datalabel datalabelattrs=(size=&FontSize) barwidth=&BarWidth nooutline;
yaxis display=(nolabel noline noticks) valueattrs=(size=&FontSize);
xaxis display=none;
run;
ods listing close; ods listing;

%mend HBARsummaryChartWithPctShares;
```

TOOL 3. SOLUTION FOR A FINITE WORK DAY IN AN ERA OF DATA OVERLOAD: SHOW THEM WHAT IS IMPORTANT . . . WITH THE MAXIMALLY INFORMATIVE SUBSETTED AND RANKED HORIZONTAL BAR CHART

For over a quarter century I have advocated and exploited use of the subsetted and ranked horizontal bar chart. I have long been committed to the idea of trying to deliver only the most important. The most important can usually fit on one sheet of paper, and frequently, if not almost always, on one web page without having to scroll. Such limitation of information volume reminds me of the advice of Kenneth J. Wesley, my staff who once counseled me, when I was agonizing over a report for executive management, that “If it doesn’t fit on one page, they won’t read it.” And I always remember the wisdom of Jim White, an expert on print document design, who said, “Let part stand for the whole.”

In that spirit, I developed a macro that allowed the user to point to a data set and create a bar chart subsetted in any one of three ways: (1) Top N (where N was any integer); (2) all values above a cut-off; or (3) enough of the top values to account for the Top P Percent of the total measure of interest. Although I have long maintained that what will fit on a page, say, anywhere from the Top 10 to the Top 40 or 50, will usually account for 80 to sometimes 99% of the total measure of interest, a reliable approach to take when subsetting is to stop reporting as soon as the chart bars account for as much of the total as you feel is important to show. YOU pick the percent target with Option 3 above. But recently I decided that I wanted a reporting system’s users to be able to optionally also look at the WHOLE list.

The macro that I am discussing in THIS section does NOT support that four-option capability. What was delivered for the client was actually a macro that not only created four versions of the ranking report, but also interlinked the four web graphs with hyperlinks—very cool and maximally convenient.

Here I provide a more limited function macro that successfully sizes the height of the image file for ANY number of the Top N bars **up to a maximum of 50** and links it forwards and backwards with an image sized large enough to display ALL of the bars. Of course, if you do not really want the ALL Bars companion chart, it is easy to modify the macro to omit that capability. (I could have built in the option, but this macro is for demonstration only.) If the Top N number of bars requested is not less than the total number of bars possible, only one web page is produced.

For this macro to perform reliably regardless of the number of Top N bars (up to 50), the SGPLOT HBAR chart image file must be sufficiently tall and/or the font used for bar labels must be sufficiently small to prevent automatic thinning of the bar labels. **See the discussion in Reference 3 of thinning of tick mark values in line plots.** At the same time, the image height must be small enough to fit in the browser window so that all of the Top N bars can be viewed without need for vertical scrolling. (The ALL Bars chart will require scrolling unless ALL is less than 51.)

So, there is a trade-off between: (a) making the bar label font small enough so that the desired number of Top N bars will not experience thinning; (b) making the bar label font small enough so that the Top N bar chart will not require vertical scrolling; and (c) making the bar label font large enough to be easily readable.

The viewing capacity of the expected target user depends on the brand and version number of the web browser, how many toolbars are in use in the web browser, and the vertical resolution of the monitor. My development and testing of the examples shown in this section were done using a monitor with 1920 X 1080 resolution, Internet Explorer Version 9, and no (what I regard as) extra toolbars.

The default font, which can be overridden by the macro user, is Albany AMT/Bold. This font is shipped with SAS software. The font size used by the macro is controlled internally, and varies based on the value of TopN. The algorithm in the macro that controls font size also determines the appropriate number of Y pixels per bar based on the value of TopN. There is an overhead number of Y pixels that is independent of the value of TopN. The overhead is a macro parameter defaulted to what I found appropriate for the default choice of bar label font. The macro dynamically determines the total number of Y pixels needed. Macro internals must be revised if using a different font for the bar labels. It required experimentation and testing to determine the relationship between TopN, font size, Y pixels per bar, and Y pixels overhead—in pursuit of preventing the dreaded automatic thinning of bar labels.

The macro default font size for the title lines is 16 pt. Reducing it might facilitate fitting your chart in a browser window on a lower resolution monitor. The title lines are OUTSIDE of and above the image file. To save vertical space, you can avoid use of the optional fifth title line. That line can be used for optional diagnostic information during testing. When not testing, you can optionally provide any text, using any color of your choice (default is black). With or without such optional text, you can include (only in black) the run day, date, and time on title line 5.

NOTE: I later provide a non-web/non-HTML version of the macro, which can create an image file on disk to be imbedded manually in Microsoft Word or Microsoft PowerPoint, and perfectly sized for whichever use you need.

Adaptive SGPLOT Horizontal Bar Chart Macro Adjusts Image Height Based On Bar Count

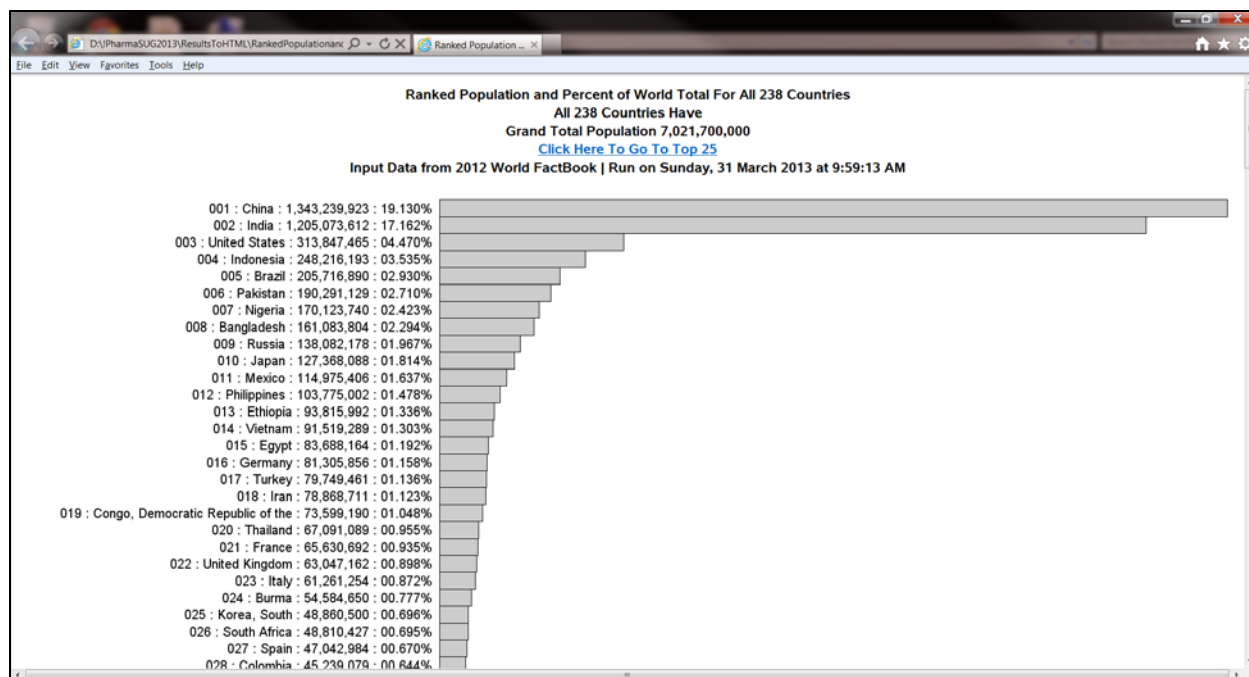
When producing a horizontal bar chart with a default size image file, software makes default use of the space. When using SAS/GRAPH PROC GCHART with a few bars, you get useless white space above and below, and with numerous enough bars you get adjacent bar label overlay. When using ODS Graphics PROC SGPLOT HBAR with a few bars, you get oversized bars to fill the space and disproportionate to the size of the bar labels, and with numerous enough bars some of the bar labels disappear (are “thinned”). The macro presented here right-sizes the image and uses SGPLOT HBAR. **It requires the CATEGORYORDER and VALUEATTRS features of Version 9.3 of SAS.**

As explained in Reference 3, I am a strong advocate of subsetting information delivered in order to focus on what's important, but I recently finally stepped up to the problem of presenting a vertically scrollable web graph of ranked horizontal bars with NO subsetting, for any number of bars, no matter how large. When you are presenting a subset of the data, it is useful to anticipate and be able to answer any question about the omitted data. The macro presented here permits you to create a pair of interlinked web graphs for a subset and for all of the data.

To deliver a tall horizontal bar chart (i.e., a very large number of bars), you obviously need to increase the pixel count for the height of the image sufficiently. To do this in a production environment, where no iterative loop of trial and adjustment is available, requires what I have long called “software intelligence”, which is my personal favorite term for data-adaptive program design and construction. The key to the solution is a macro that dynamically sets the needed pixel count capacity for the height of the image based on bar count and dynamically sets the font size.

My later examples use the SASHELP.SHOES data set that every SAS user can access, but I first want to use data that involves a larger number of potential bars—i.e., population by country data from the 2012 World FactBook.

NOTE: TITLE5 is optional: the descriptive part is specified with an optional macro parameter, and the run day, date, & time can be omitted. Alternatively TITLE5 can instead deliver diagnostic information—e.g., if desired during testing.

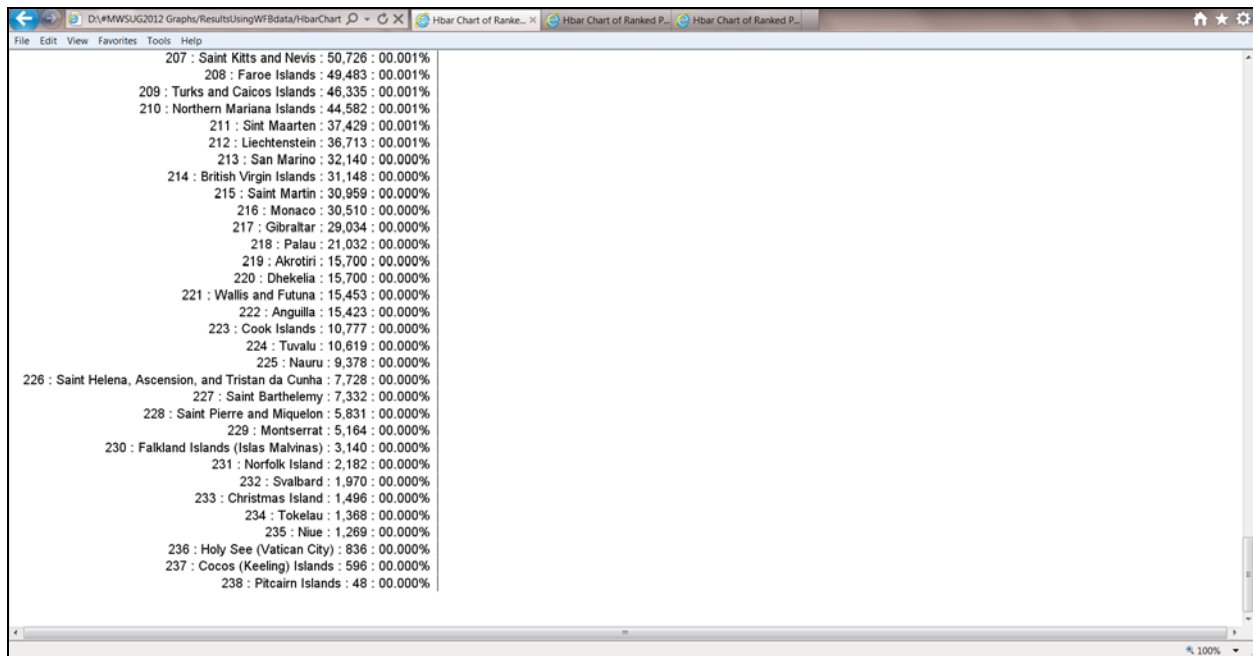


Here is an enlargement of the top four title lines:

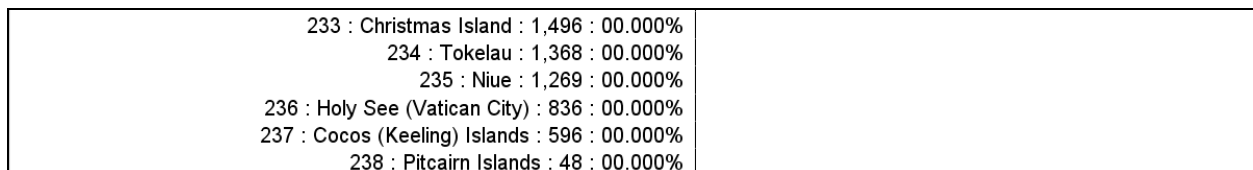
Ranked Population and Percent of World Total For All 238 Countries
All 238 Countries Have
Grand Total Population 7,021,700,000
[Click Here To Go To Top 25](#)

The count of bars and the grand total of the measure of interest are dynamically generated by the macro processing. Note the link to a Top 25 subset of the data. That web page has a link back.

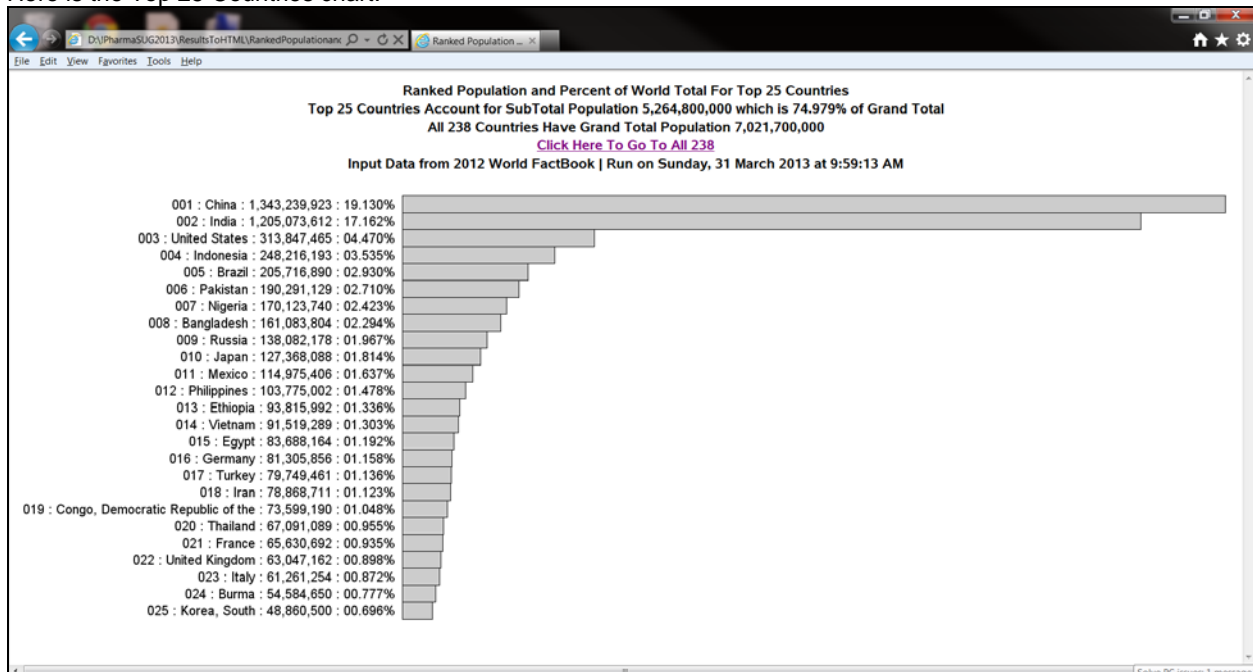
Here is the bottom of the All 238 Countries chart:



In the enlargement below, the bottom bars are nothing but a vertical line at this point, and percents of the whole, even to three decimal places, are 00.000%, but the country names and populations are still useful information.



Here is the Top 25 Countries chart:



Here is an excerpt of the chart, showing only the title, subtitles, and the link back to the All 238 Countries chart:

<p>Ranked Population and Percent of World Total For Top 25 Countries Top 25 Countries Account for SubTotal Population 5,264,800,000 which is 74.979% of Grand Total All 238 Countries Have Grand Total Population 7,021,700,000 Click Here To Go To All 238</p>
--

The first subtitle provides the bar count, the subtotal from those bars, and their percent of the grand total. The second subtitle provides both the total bar count and their grand total. Both subtitles are dynamically generated by the macro.

The code was used to create web graphs for display on a wide-screen monitor with resolution 1920 X 1080. During initial testing with Xpixels=1875 to make full use of the browser window, PROC SGPLOT failed with this message:

ERROR: Java virtual machine exception. java.lang.OutOfMemoryError: Java heap space.

The problem could be avoided by reducing the chart image file width with Xpixels=1676. However, by doing some investigation at support.sas.com, by use of a search for "java heap space" in the Samples and SAS Notes (on web page <http://support.sas.com/notes/index.html>), I found <http://support.sas.com/kb/31/184.html> for which the title is 'Problem Note 31184: "Java virtual machine exception" message issued with ODS Graphics and large data sets'. My example DOES NOT involve a "large data set", but it does create a somewhat large image file. (Of course, what constitutes "large" is inherently ambiguous.) The recommended solution in the Problem Note is to make a change to the SASV9.CFG file. If you are using SAS on a standalone PC, you should be able to make the change. If using SAS on a server, you presumably will need to request the change to be made by the SAS server administrator.

NOTE: Instead of actually encountering the ERROR above, for some large image files, you might instead find this message in your SAS log:

WARNING: A very large output size of (1875, 6219) is in effect. This could make Java VM run out of memory and result in some Java exceptions. You should reduce the output size or DPI settings.

where the width and height pixel counts, here 1875 and 6219 respectively, will be replaced by the values appropriate to YOUR graph. These numbers are determined by parameters on the ODS GRAPHICS statement, namely:

width=1875px height=6219px

where 1875 comes from a macro invocation parameter and 6219 is dynamically set by the macro logic.

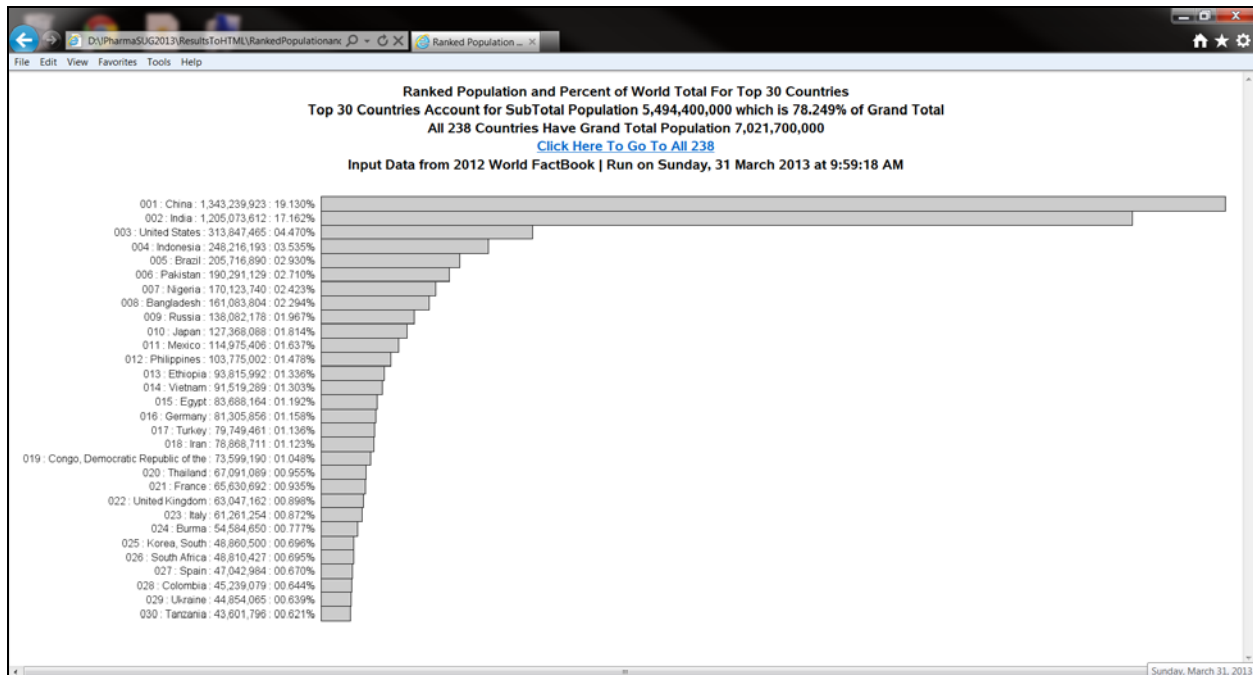
Here is the macro invocation code used to create the two web-interlinked charts shown above:

```
LIBNAME DataLib "D:\!PharmaSUG2013\Data";
%SubsettedRankingHbarChartsToHTML(
data=DataLib.PopulationByCountryPerWFB2012
,TopN=25
,RptPath=D:\!PharmaSUG2013\ResultsToHTML
,Title1_Prefix=Ranked Population and Percent of World Total For
,Title1_Suffix=Countries
,Title5=Input Data from 2012 World FactBook /* specifying Title5 is optional */
/* ,RunDayDateTimeInTitle5=NO */ /* Uncomment prior to turn off macro default YES */
,BarLabelVar=Country
,BarMeasureVar=Population
/* ,BarMeasureDescription=Population AsOf 2012 WFB Estimate */
/* With the above commented out, chart titles use BarLabelVar value as description */
,BarMeasureFormat=comma13.
,Xpixels=1875); /* requires a 1920 pixel width monitor */
```

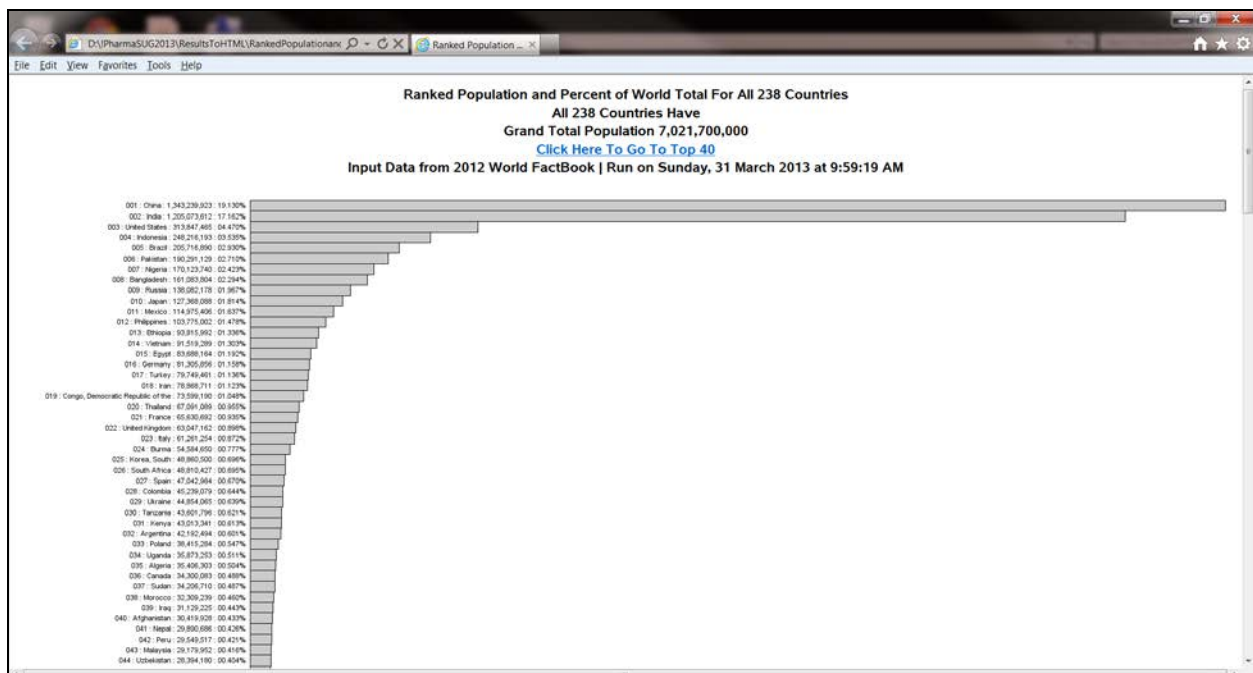
Every macro parameter shown above must be assigned, except Title5 which defaults to null. It is important to realize that the macro provided in this paper does NOT verify that you have assigned all of the mandatory parameters, or that you have provided appropriate values. It does, however, make some checks, and, if you do not override the macro default selection for TextFont, it selects a point size for the bar labels small enough to prevent thinning by PROC SGPLOT, but not smaller than necessary. In Reference 3, I discussed and demonstrated the

unfortunate ODS Graphics / SG Procedure phenomenon of thinning of axis tick mark values, but there in the context of line charts, not bar charts. Remember that thinning occurs without any WARNING message in the SAS log. If you use a different font for the bar labels, you might need to change macro internal code and the **YpixelsOverhead** macro parameter (which is defaulted to 31) in order to prevent thinning of bar labels. For values of TopN greater than 25, the macro uses progressively smaller point sizes for the TextFont, as shown in the examples below. Values of TopN greater than 50 are infeasible if you want readable labels and to keep all of the bars within the web browser window without scrolling. The macro rejects use of infeasible values for TopN.

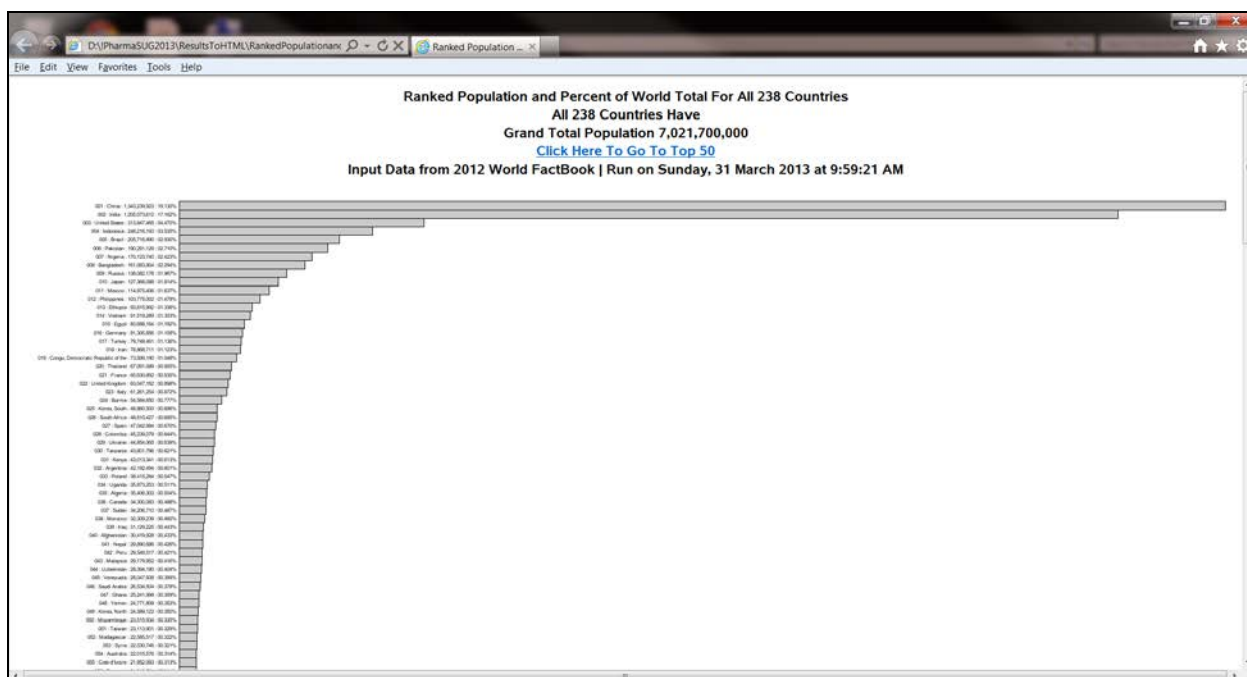
Below is a Top 30 chart. The only macro invocation code change was to set TopN=30.



When shrunk below to fit within 6.5 inches for this paper, the Top 40 chart might, with some difficulty, be readable, especially if you are already familiar with Country names. The digits on the bar labels are even harder to read.



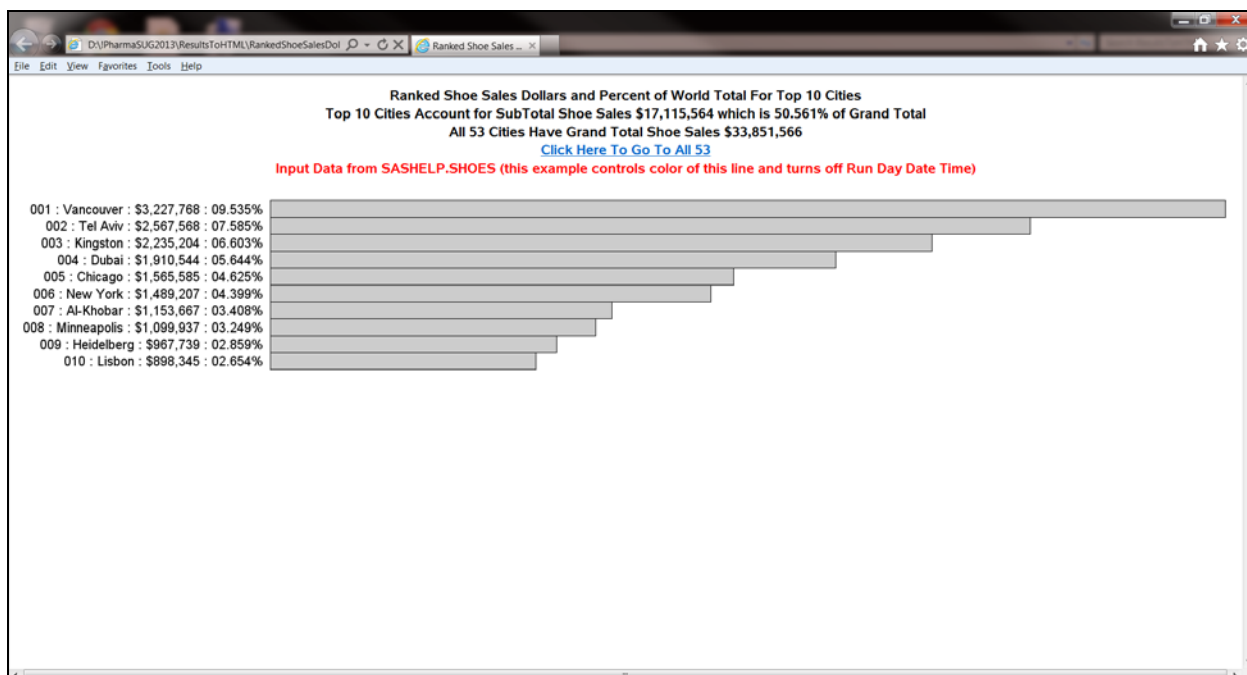
The Top 50 chart was easily readable on the 17-inch diagonal monitor used for development, which has a screen width of 15 inches. Here, shrunk below to fit within 6.5 inches for this paper, the bar labels are unreadable.



The SAS-Institute-developed software-internal code does not adjust the font size. This macro does that.

NOTE: A macro presented later as Tool 5 turns on ALT text to display the bar label information in a pop-up box when the mouse is rested on any bar when the macro knows that the bar label text is a small font size.

Below is an example for TopN=10, using the SASHELP.SHOES data set that any SAS user can access. The bars move up the browser window because the macro assigns a shorter height to the image file. Without the macro, the software itself would work with whatever height that is assigned, if any height IS assigned, on the ODS GRAPHICS statement, or with its own default of 480 pixels, and would center the bars vertically within that image file, making the bars as wide as possible in the available space, even if their width would be unnecessarily large and out of proportion to the bar label height. Such results will be shown later in the paper.



Here is the macro invocation code used to create the Top 10 Chart:

```
%SubsettedRankingHbarChartsToHTML(
data=SASHELP.SHOES
,TopN=10
,RptPath=D:\!PharmaSUG2013\ResultsToHTML
,Title1_Prefix=Ranked Shoe Sales Dollars and Percent of World Total For
,Title1_Suffix=Cities
,Title5=Input Data from SASHELP.SHOES (this example controls color of this line and
turns off Run Day Date Time)
,Title5_Color=red
,RunDayDateTimeInTitle5=NO
,BarLabelVar=Subsidiary
,BarMeasureVar=Sales
,BarMeasureDescription=Shoe Sales
,BarMeasureFormat=dollar11.
,Xpixels=1875); /* requires a 1920 pixel width monitor */
```

Unless you override the macro's default choices for its TextFont or YpixelsOverhead parameters, the SubsettedRankingHbarChartsToHTML macro will deliver your output graph with no risk of missing bar labels due to thinning by the SAS software. There is a limit of 50 on TopN, but, above that, the font size needed to avoid thinning would have to be so small as to be unreadable.

NOTE: If the expected target user monitor has a vertical resolution lower than 1080 pixels, the macro internals and YpixelsOverhead value will need to be revised if you wish the entire bar chart, up to a certain maximum bar count (unique to the monitor resolution), to be viewable without vertical scrolling—which is an important design objective.

Here is the code for the SubsettedRankingHbarChartsToHTML macro:

```
%macro SubsettedRankingHbarChartsToHTML(
data=
,RptPath=
,Title1_Prefix=
,Title1_Suffix=
,UseTitle5ForDiagnostics=NO /* YES could be useful during testing a revised macro */
,Title5= /* Title5 is optional */
,Title5_Color=black
,RunDayDateTimeInTitle5=YES /* If using TITLE5 for diagnostic
or for a custom message,
it might be wise to turn off Run Day, Date, & Time
to prevent wrap of a too long title line, which could
push the bottom of chart below the browser window. */
,TitlesJustify=Center /* Center is actually the SAS default.
If titles were imbedded in the graph,
SGPLOT HBAR chart centering of titles would be defective.
Left Justification would be the better choice.
ODS HTML NOGTITLE (used below in macro) makes Center OK. */
,TopN= /* subset number of bars to be shown */
,BarLabelVar=
,BarMeasureVar=
,BarMeasureFormat=
,BarMeasureDescription= /* Macro does not retrieve SAS var label even if one exists.
If BarMeasureDescription not specified,
then SAS var name (BarMeasureVar) is used instead. */
,BarColor=CXCCCCCC /* light grey */
,TitleFont='Albany AMT/Bold'
,TitleFontPointSize=16
,TextFont='Albany AMT/Bold'
,Xpixels= /* 1875 is maximum to avoid horizontal scrolling on 1920-pixel monitor.
Select a value based on the expected viewing monitor.
REDUCE Xpixels if java heap space out of memory & SGPLOT failure
when bar count of the All Bars companion chart is high.
The memory problem is announced with these messages in the SAS log:
```

```

        ERROR: Java virtual machine exception.
            java.lang.OutOfMemoryError: Java heap space.
        NOTE: The SAS System stopped processing this step because of errors.
        Alternatively, you might get relief by changing the SASV9.CFG file,
        following the instructions
        in SAS Note http://support.sas.com/kb/31/184.html
        If you are using SAS on a server, you will need to request the change
        be made by the server SAS Administrator. */
,YpixelsOverhead=31 /* 31 YpixelsOverhead is for use with TextFont='Albany AMT/Bold'
        and with the rule inside the macro for choosing
        TextFontPointSize and YpixelsPerBar based on the value of TopN.
        YpixelsOverhead, YpixelsPerBar, TopN are used inside the macro
        to compute the total Y pixels for height of the image file. */
);

%if %upcase(&UseTitle5ForDiagnostics) EQ YES AND %length(&Title5) NE 0 %then %do;
    %put USER ERROR: With the &sysmacroname macro, if you specify a value for Title5,
    you must specify UseTitle5ForDiagnostics=NO;
    %GoTo MacExit;
%end;

%if "%upcase(&TextFont)" NE "'ALBANY AMT/BOLD'" %then %do;
    %put USER ERROR: This &sysmacroname macro is currently configured to only work with
    the Albany AMT/Bold as TextFont parameter;
    %GoTo MacExit;
%end;

%if %length(&BarMeasureDescription) EQ 0
%then %let BarMeasureDescription = &BarMeasureVar;

%if %eval(&TopN LE 25)
%then %do;
    %let TextFontPointSize = 14;
    %let YpixelsPerBar = 26;
%end;
%else
%if %eval(&TopN LE 30)
%then %do;
    %let TextFontPointSize = 12;
    %let YpixelsPerBar = 21.67;
%end;
%else
%if %eval(&TopN LE 40)
%then %do;
    %let TextFontPointSize = 8;
    %let YpixelsPerBar = 16.25;
%end;
%else
%if %eval(&TopN LE 50)
%then %do;
    %let TextFontPointSize = 6;
    %let YpixelsPerBar = 13;
%end;
%else %do;
    %put USER ERROR: This &sysmacroname macro is currently configured to only work with
    values less than or equal to 50 for the TopN parameter;
    %GoTo MacExit;
%end;

DATA _NULL_;
RunDayDateTimeText = 'Run on ' || TRIM(LEFT(PUT(DATE(),weekdatx37.))) || ' at ' ||
    TRIM(LEFT(PUT(TIME(),timeampm11.)));
CALL SYMPUT('RunDayDateTime',TRIM(LEFT(RunDayDateTimeText)));

```

```

RUN;

PROC SUMMARY DATA=&data nway;
CLASS &BarLabelVar;
VAR &BarMeasureVar;
OUTPUT OUT=Summed(drop=_type_ _freq_) SUM=;
RUN;

PROC SQL NOPRINT;
SELECT COUNT(&BarLabelVar) , SUM(&BarMeasureVar)
      INTO :CountOfAllBars      , :BarMeasureGrandTotal
      FROM Summed;
QUIT;

DATA _NULL_;
LENGTH ForSYMPUT 8;
ForSYMPUT = &BarMeasureGrandTotal;
CALL SYMPUT('GrandTotal',TRIM(LEFT(PUT(ForSYMPUT,&BarMeasureFormat))));
RUN;

PROC SORT DATA=Summed;
BY DESCENDING &BarMeasureVar;
RUN;

DATA ALLtoChart_WithBarLabels(drop=BarLabelSuffix);
LENGTH BarLabel $ 256 BarLabelSuffix $ 7;
SET Summed;
BarLabelSuffix =
      TRIM(LEFT(PUT(((&BarMeasureVar / &BarMeasureGrandTotal) * 100),6.3))) || '%';
IF LENGTH(BarLabelSuffix) EQ 6
THEN BarLabelSuffix = '0' || BarLabelSuffix;
BarLabel = TRIM(LEFT(PUT(_N_,z3.))) || ' : ' || /* assumes max possible is 999 bars */
      TRIM(LEFT(&BarLabelVar)) || ' : ' ||
      TRIM(LEFT(PUT(&BarMeasureVar,&BarMeasureFormat))) || ' : ' || BarLabelSuffix;
RUN;

OPTIONS OBS=&TopN;

DATA TOPNtoChart_WithBarLabels;
SET ALLtoChart_WithBarLabels;
RUN;

OPTIONS OBS=MAX;

%macro TopN_OR_All(Which=);

%let Which = %upcase(&Which);

%if &Which EQ TOPN %then %do;

PROC SQL NOPRINT;
SELECT SUM(&BarMeasureVar)
      INTO :BarMeasureSubTotal
      FROM TopNtoChart_WithBarLabels;
QUIT;

DATA _NULL_;
LENGTH ForSYMPUT1 ForSYMPUT2 8;
ForSYMPUT = &BarMeasureSubTotal;
CALL SYMPUT('SubTotal',TRIM(LEFT(PUT(ForSYMPUT,&BarMeasureFormat))));
ForSYMPUT1 = &BarMeasureSubTotal;
ForSYMPUT2 = &BarMeasureGrandTotal;
CALL SYMPUT('SubTotalPercentOfGrandTotal',

```

```

    TRIM(LEFT(PUT(( ForSYMPUT1 / ForSYMPUT2 ) * 100 , 6.3))));
RUN;

%end;

%if &Which EQ TOPN
%then %let CountOfSelectedBars = &TopN;
%else %let CountOfSelectedBars = &CountOfAllBars;

%let Y_pixels = %eval( %sysfunc(CEIL( %sysevalf(&CountOfSelectedBars * &YpixelsPerBar)
)) + &YpixelsOverhead );

%if &Which EQ ALL
%then %do;
    %let Title1 = &Title1_Prefix.%str( All )&CountOfAllBars &Title1_Suffix;
    %let Title2 = All &CountOfAllBars &Title1_Suffix Have;
    %let Title3 = Grand Total &BarMeasureDescription &GrandTotal;
    %let Title4_LinkDescription = Top &TopN;
%end;
%else %do;
    %let Title1 = &Title1_Prefix.%str( Top )&TopN &Title1_Suffix;
    %let Title2 = Top &CountOfSelectedBars &Title1_Suffix
        Account for SubTotal &BarMeasureDescription &SubTotal
        which is &SubTotalPercentOfGrandTotal% of Grand Total;
    /* Because Title2 is delivered by HTML source code
       due to the specification of ODS HTML . . . NOGTITLE
       any extra blanks between the three parts of Title2
       on three separate lines in the %LET statement above
       will be compressed down to one blank. */
    %let Title3 =
        All &CountOfAllBars &Title1_Suffix Have Grand Total &BarMeasureDescription
&GrandTotal;
    %let Title4_LinkDescription = All &CountOfAllBars;
%end;

%if %eval(&CountOfAllBars GT &TopN) %then %do;

    %let FileName_Temp = %sysfunc(COMPRESS(&Title1,' '));
    %let FileNameLinkTo_Temp =
%sysfunc(COMPRESS(&Title1_Prefix.&Title4_LinkDescription&Title1_Suffix));

    %let FileName = &FileName_Temp._LinkedTo_&FileNameLinkTo_Temp;
    %let FileNameLinkTo = &FileNameLinkTo_Temp._LinkedTo_&FileName_Temp;

%end;
%else %let FileName = %sysfunc(COMPRESS(&Title1,' '));

%if &UseTitle5ForDiagnostics EQ YES %then %do;
    %let Title5ForDiagnostics =
        &CountOfSelectedBars bars, &YpixelsPerBar Ypixels per bar,
        &YpixelsOverhead Ypixels overhead, &Y_pixels Total Ypixels, &Xpixels Xpixels,
        Titles &TitleFontPointSize.pt &TitleFont, Bar Labels &TextFontPointSize.pt
        &TextFont;
    %let Title5ForDiagnostics = %sysfunc(compbl(%quote(&Title5ForDiagnostics)));
%end;

ODS GRAPHICS ON / RESET=ALL BORDER=OFF SCALE=OFF
HEIGHT=&Y_pixels.px WIDTH=&Xpixels.px
IMAGENAME="&FileName";

FOOTNOTE;
ODS NORESULTS;
ODS LISTING CLOSE;

```



```

ODS HTML PATH="&RptPath" (URL=NONE) STYLE=Styles.MinimalWithNoFrame
NOGTITLE NOGFOOTNOTE
BODY="&FileName..html" (TITLE="&Title1");

TITLE1 JUSTIFY=&TitlesJustify FONT=&TitleFont HEIGHT=&TitleFontPointSize PT "&Title1";
TITLE2 JUSTIFY=&TitlesJustify FONT=&TitleFont HEIGHT=&TitleFontPointSize PT "&Title2";
TITLE3 JUSTIFY=&TitlesJustify FONT=&TitleFont HEIGHT=&TitleFontPointSize PT "&Title3";
TITLE4 JUSTIFY=&TitlesJustify FONT=&TitleFont HEIGHT=&TitleFontPointSize PT
%if %eval(&CountOfAllBars GT &TopN)
%then %do;
    LINK="&FileNameLinkTo..html" "Click Here To Go To &Title4_LinkDescription";
%end;
%else %do;
    "You requested the Top &TopN &BarMeasureDescription., but there are only
    &CountOfAllBars &Title1_Suffix";
%end;
%if %upcase(&UseTitle5ForDiagnostics) EQ YES
    or
    %length(&Title5) NE 0
    or
    %upcase(&RunDayDateTimeInTitle5) EQ YES
%then %do;
TITLE5 JUSTIFY=&TitlesJustify FONT=&TitleFont HEIGHT=&TitleFontPointSize PT
    %if %upcase(&UseTitle5ForDiagnostics) EQ YES %then %do;
        COLOR=red HEIGHT=14 PT /* make this long text string smaller */
        "&Title5ForDiagnostics "
    %end;
    %else
        %if %length(&Title5) NE 0 %then %do;
            COLOR=&Title5_Color
            "&Title5 "
            COLOR=black
        %end;
        %if %upcase(&RunDayDateTimeInTitle5) EQ YES
        %then %do;
            "| &RunDayDateTime"
        %end;
    ;
%end;

PROC SGPLOT DATA=&Which.toChart_WithBarLabels;
HBAR BarLabel / RESPONSE=&BarMeasureVar
    CATEGORYORDER=RESPDESC BARWIDTH=1 FILL FILLATTRS=(COLOR=&BarColor) OUTLINE;
YAXIS DISPLAY=(NOLABEL NOLINE NOTICKS) VALUEATTRS=(SIZE=&TextFontPointSize PT);
XAXIS DISPLAY=NONE;
RUN;
ODS HTML CLOSE;
ODS LISTING;

%mend TopN_OR_All;

%macro RemoveCraphAreaFrameFromStyle
(ParentStyle=,StyleWithNoFrame=);
PROC TEMPLATE;
DEFINE STYLE &StyleWithNoFrame;
    PARENT=&ParentStyle;
    CLASS GRAPHWALLS / FRAMEBORDER=OFF; /* remove a useless box around the bars */
END; RUN;
%mend RemoveCraphAreaFrameFromStyle;

%RemoveCraphAreaFrameFromStyle
(ParentStyle=Styles.Minimal,StyleWithNoFrame=Styles.MinimalWithNoFrame);

```

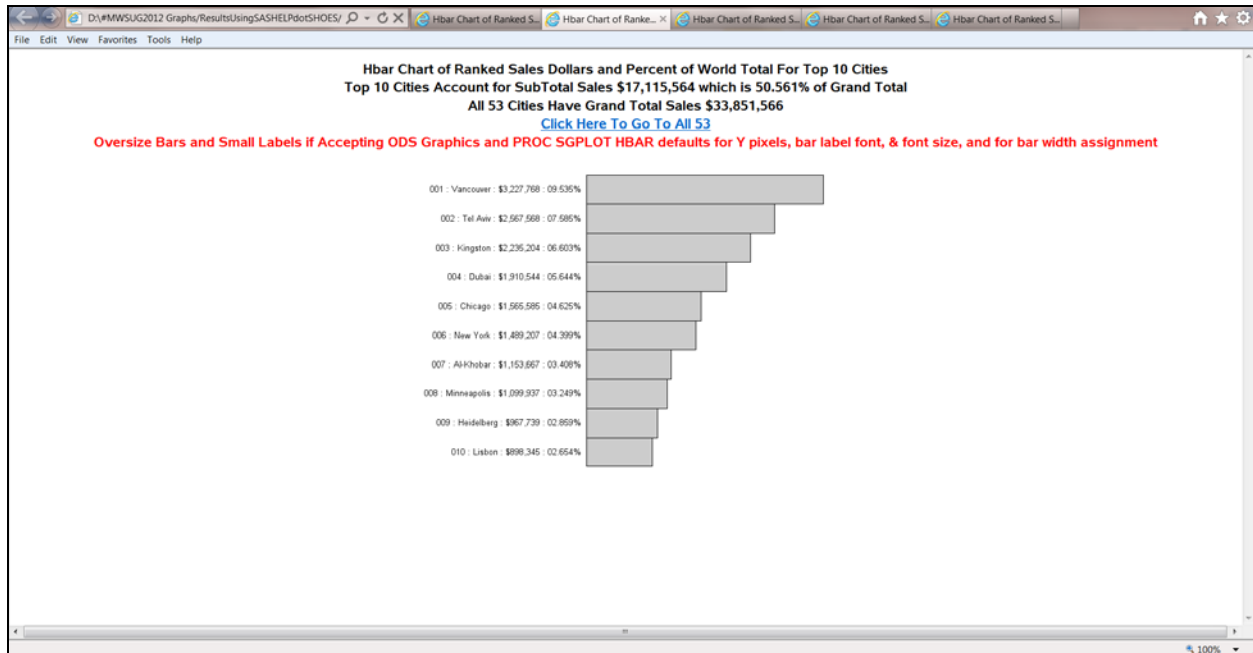
```

%TopN_OR_All(Which=All);
%if %eval(&CountOfAllBars GT &TopN) %then %do;
  %TopN_OR_All(Which=TopN);
%end;

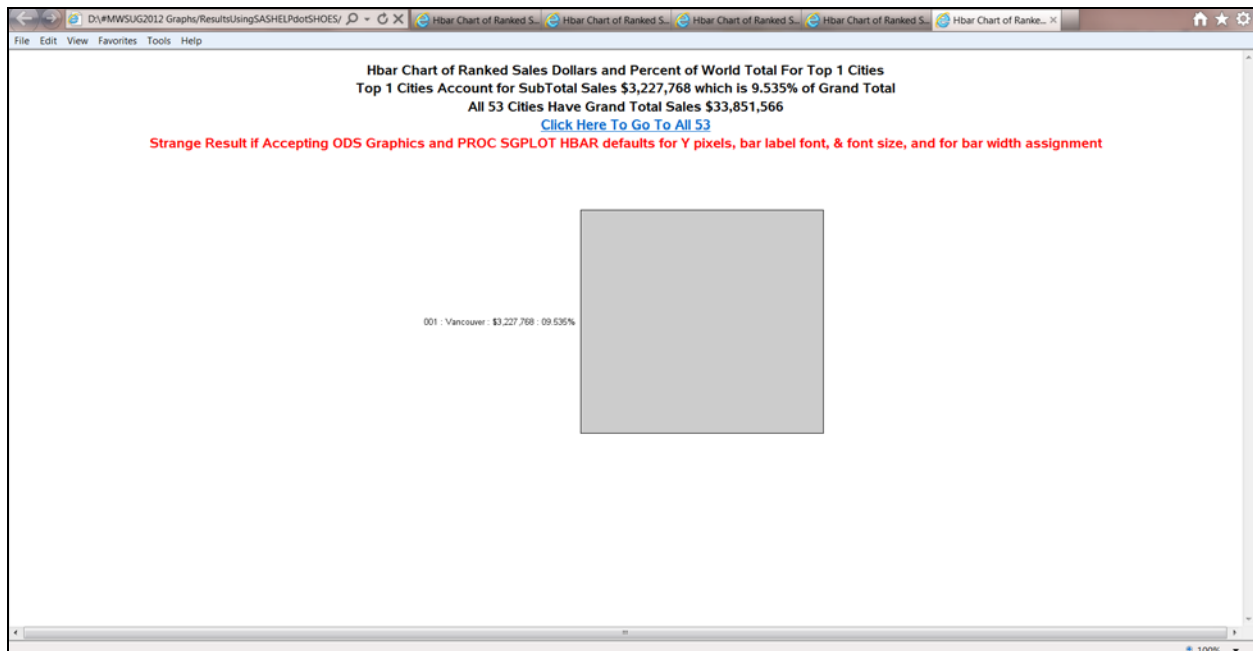
%MacExit:
%mend SubsettedRankingHbarChartsToHTML;

```

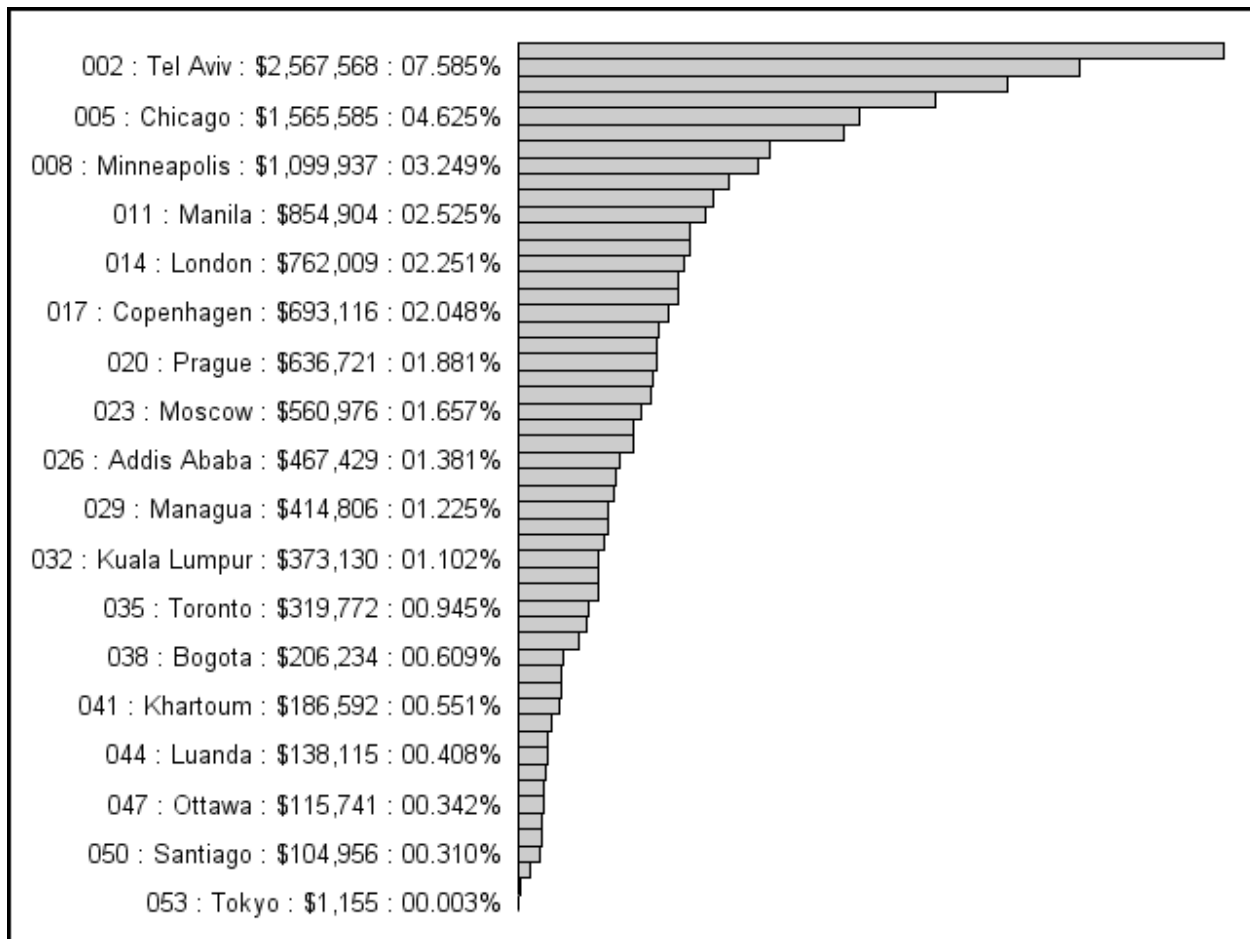
Below is a demonstration of what happens if you let ODS Graphics and PROC SGPLOT “do what comes naturally”—i.e., if you remove the built-in data-adaptive software intelligence from the macro and do not use the macro’s built-in controls for image height sizing and font sizing to create a Top 10 chart. You get wider bars and disproportionately sized bar labels.



To dramatize the problem further, below is the result for a Top 1 chart, NOT using the macro’s built-in controls.



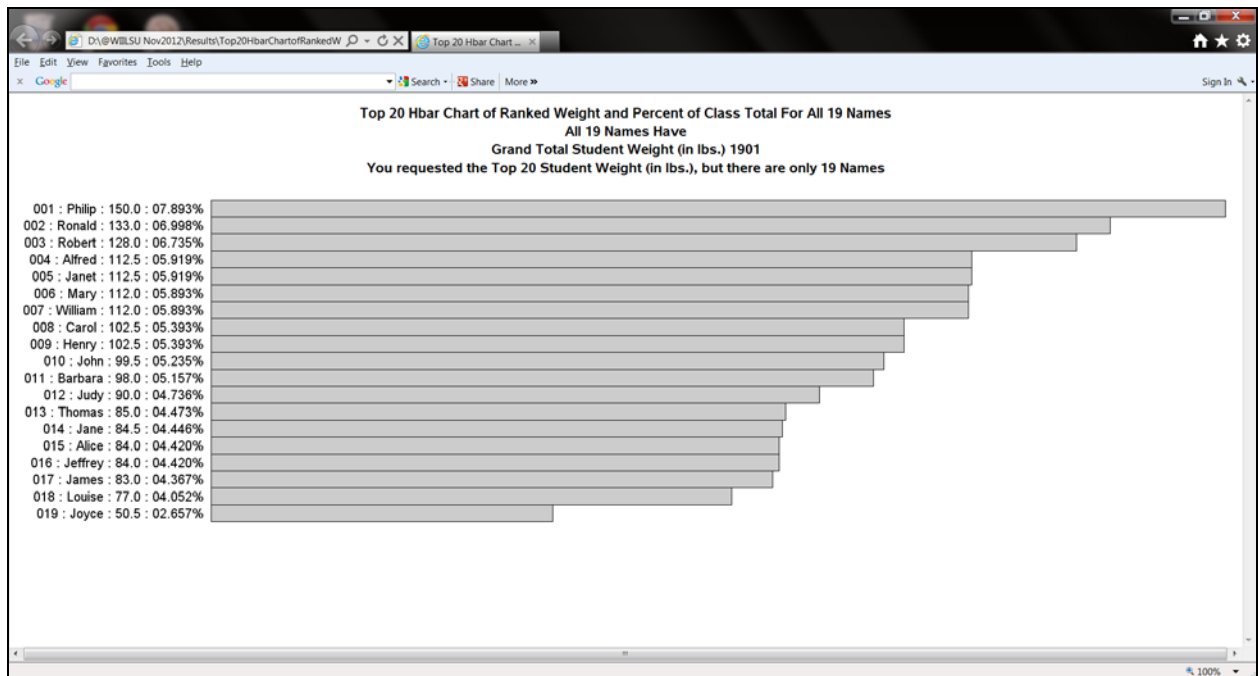
It is instructive to also see the **unusable** All 53 Cities chart which is the linked companion of the charts in the web pages above. Inserted below is the image file (shrunk upon manual insert here to fit on this page), not a screen capture of the web page that imbeds it.



ODS Graphics PROC SGPLOT inflicts the above thinning of the bar labels with no mention in the SAS log.

The undesirable results shown above from using defaults for a Top 10 chart, a Top 1 chart, and an All Bars chart can be overcome by manually adding, changing, and tuning some additional ODS Graphics and/or PROC SGPLOT HBAR options. The purpose of a macro solution alternative, which does all of those adjustments for you, is to allow your unchanging production job program, or your written-one-time program invoked by a real-time request, to simply identify the data set, the variables, a SAS format for the measurement variable, optionally a description for the measurement variable (to be used in titles instead of the measurement variable name), a title prefix, a title suffix to provide a title-appropriate label for the category variable, the value for TopN, and a value appropriate for the pixel width of the expected target monitor, and to **get the result right the first time every time.**

Before closing this section, on the following page is a demonstration of what happens, IF USING the macro, when requesting a subset TopN that is equal to or exceeds the maximum number of possible bars. Note the message on title line 4. Rather than failing or rejecting the request, the macro is forgiving and informative.



TOOL 4. Right-Sized Subsetted Ranking Hbar Chart For PowerPoint Slides Or Word Documents

The macro provided in this section is intended to provide all of the automatic data summarization, data management, layout formatting, and dynamic highly informative titling done by the macro presented in the prior section to produce interlinked web charts, but instead produces only one chart, the subsetted version, and allows you to easily control the dimensions. The examples below use pixel dimensions appropriate to fill a PowerPoint slide and to fill a portrait-oriented American Letter size page with one-inch margins.

In the case of PowerPoint slides, it might be a convenience to include all possible text related to the image inside the image itself, rather than having to provide the text with PowerPoint on the slide after (or before) inserting image. However, it is not difficult to build the slide in two parts, and keeping some text in the slide does permit one to easily change the text without rerunning the image creation. To leave some vertical space for on-slide-provided text simply requires adjusting the image height parameter when invoking the macro. The pixel count used in the examples here for full-height slides assumes that the physical height of the slide is 7.5 inches. Based on how much vertical space you want to be able to use for on-slide text, you need to simply reduce your Y pixel count proportionately.

In the case of a Word document, you would not want to fill the page unless the graph is very dense and will not communicate well in a smaller "window". To deliver it in a smaller size for Word likewise simply means scaling down the example parameters used below. Also, if you do want to present it in a full-size landscape page rather than portrait, you only need to switch your use of the example Y pixel and X pixel counts.

Below is the code to create a graph to fit on a full 8.5-inch X 11-inch page with 1-inch margins in a Microsoft Word document. The source document for this paper has such characteristics.

NOTE: If I had omitted the optional Title4 below,
I could have increased the TextFontPointSize to 8 without causing thinning of bar labels.

```
LIBNAME DataLib "D:\!PharmaSUG2013\Data";

options MPRINT;

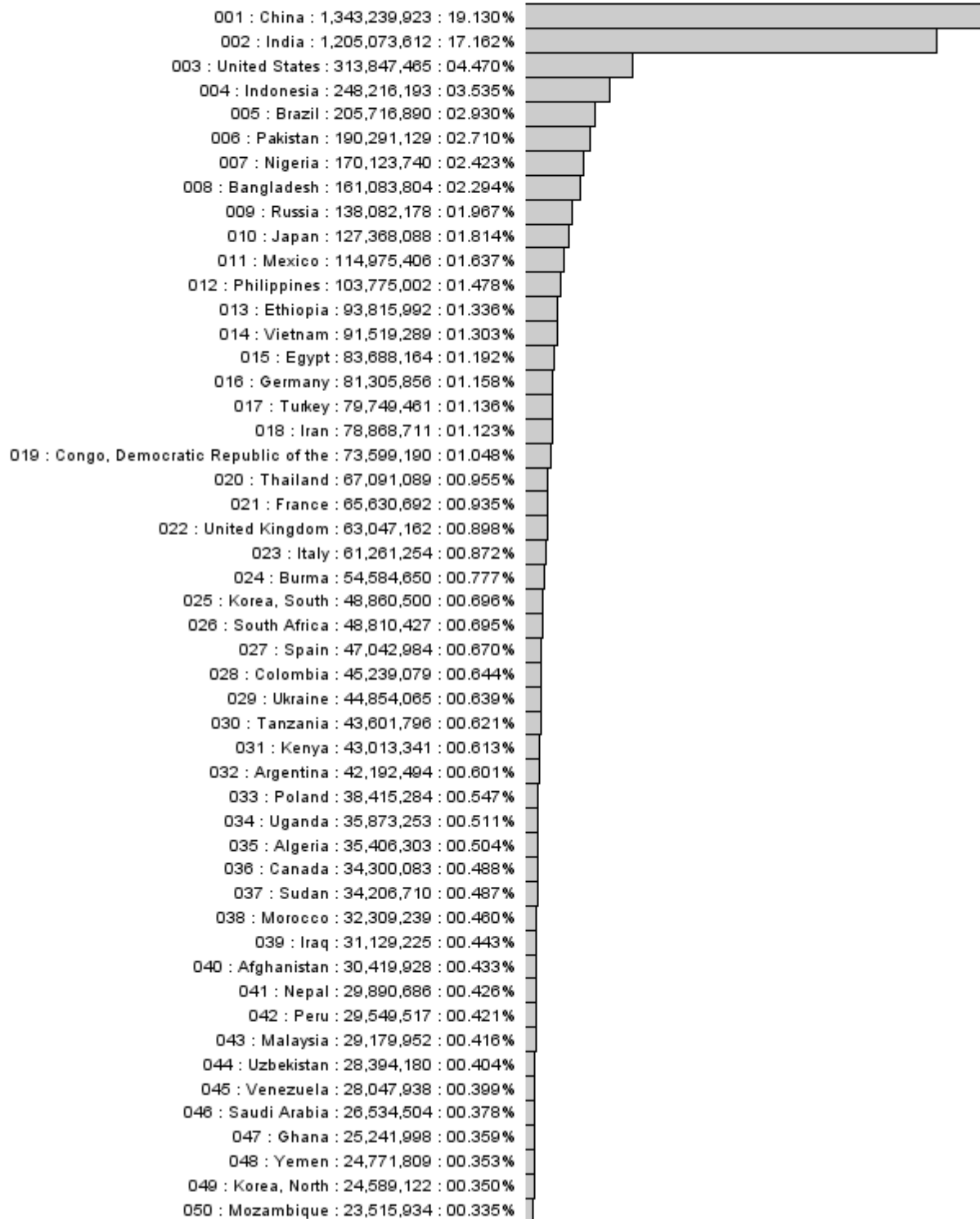
%SubsettedRankingHbarChartsToDisk(
data=DataLib.PopulationByCountryPerWFB2012
,TopN=50
,OutputDiskPath=D:\!PharmaSUG2013\ResultsForManualInsertToMicrosoftOffice
,GraphFileName=ImageToInsertIntoAMicrosoftWordPage
,BorderOnOrOff=ON
,Title1_Prefix=MS Word Hbar Chart of Ranked Population and Percent of World Total For
,Title1_Suffix=Countries
,Title4=Increasing font point size would have caused thinning of labels
,Title4_Color=Red
,ShowRunDayDateTime=YES
,BarLabelVar=Country
,BarMeasureVar=Population
,BarMeasureFormat=comma13.
,TitleFontPointSize=9 /* adjusted to avoid TITLE line wrap
                        (no WARNING in SAS log when they occur) */
,TextFontPointSize=7 /* adjusted to avoid thinning of bar labels
                      (no WARNING in SAS log when it occurs) */
,Xpixels=624 /* image width to fill 6.5 inches in Microsoft Word document */
,Ypixels=864); /* image height to fill 9 inches in Microsoft Word document */
```

On the next page, the image file was imbedded by using the Word Insert Picture function. I applied a Word border in addition to the chart's own border. In order to get the bottom border to appear at its full thickness, it was necessary to use the Word Size function to reduce the imbedded image height to 8.99 inches.

MS Word Hbar Chart of Ranked Population and Percent of World Total For Top 50 Countries
Top 50 Countries Account for SubTotal Population 6,121,300,000 which is 87.177% of Grand Total
All 238 Countries Have Grand Total Population 7,021,700,000

Increasing font point size would have caused thinning of labels

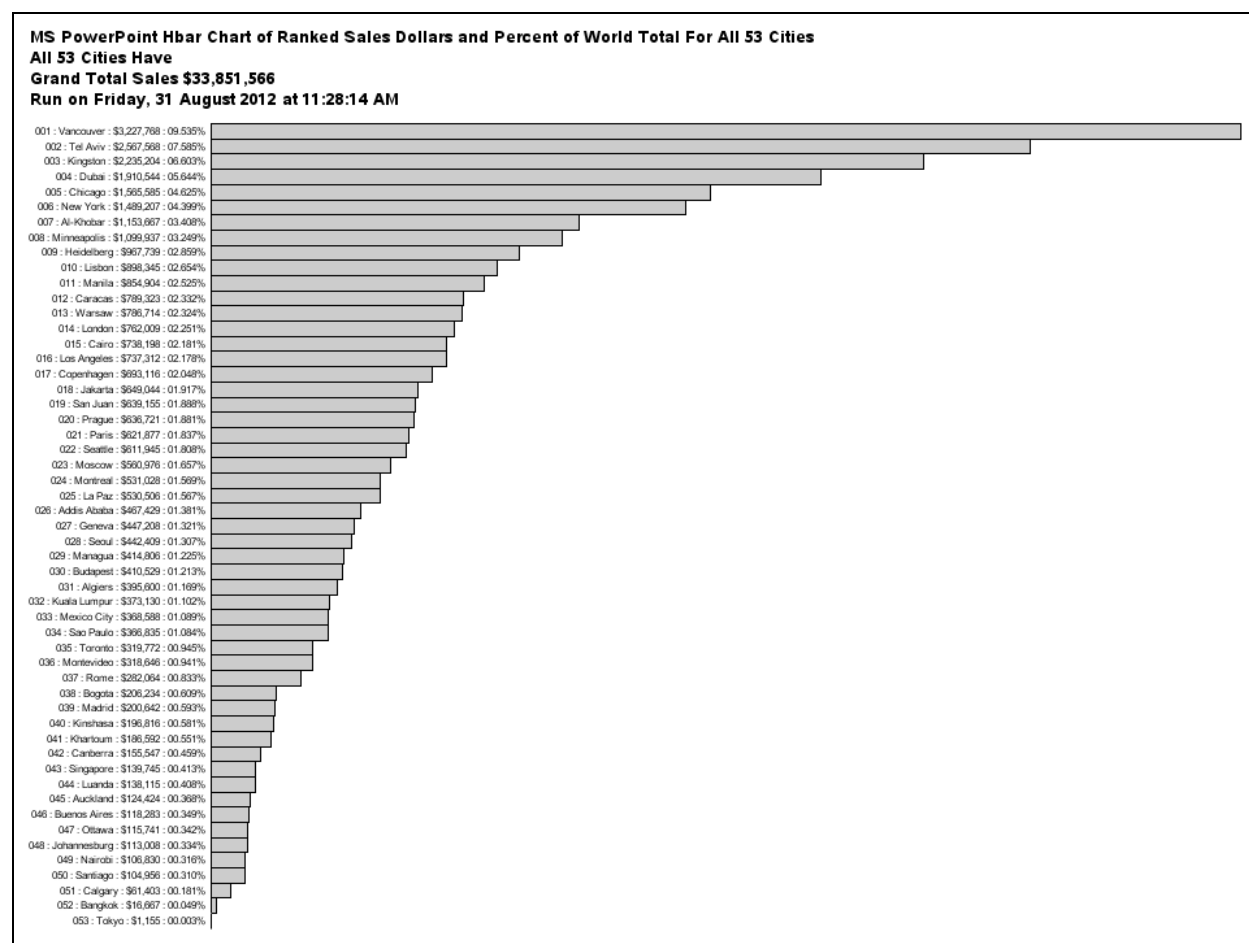
Run on Friday, 31 August 2012 at 11:28:13 AM



Below is the code to create a graph to fit on the full 10-inch by 7.5-inch area of a Microsoft PowerPoint slide. Unlike the prior example for a graph imbedded in a Word document, I did not turn on the optional graph border, which cannot be visible on an image that fills the slide. Also, to save vertical space, this derivative macro does not include the option of Title5.

```
%SubsettedRankingHbarChartsToDisk(
data=sashelp.shoes
,TopN=999999999 /* I wanted a chart of ALL of the bars. */
/* Any number greater than the total number of bars forces that. */
,OutputDiskPath=D:\!PharmaSUG2013\ResultsForManualInsertToMicrosoftOffice
,GraphFileName=ImageToInsertIntoAMicrosoftPowerPointSlide
,Title1_Prefix=MS PowerPoint Hbar Chart of Ranked Sales Dollars and Percent of World
Total For
,Title1_Suffix=Cities
,BarLabelVar=Subsidiary
,BarMeasureVar=Sales
,BarMeasureFormat=dollar11.
,TitleFontPointSize=10 /* adjusted to avoid thinning of bar labels or TITLE line wrap
(no WARNING in SAS log when they occur) */
,TextFontPointSize=6 /* adjusted to avoid thinning of bar labels
(no WARNING in SAS log when it occurs) */
,Xpixels=960 /* image width to fill 10 inches in Microsoft PowerPoint slide */
,Ypixels=720); /* image height to fill 7.5 inches in Microsoft PowerPoint slide */
```

Below is the image created by running the above code. Its readability here is impaired by the forced shrinkage to fit within the 6.5-inch width of the usable part of this page.



Here is the code for the SubsettedRankingHbarChartsToDisk macro:

```
%macro SubsettedRankingHbarChartsToDisk(
data=
,OutputDiskPath=
,ShowRunDayDateTime=NO /* It appears as Title5 if Title4 is used.
                        Otherwise, as Title4 */
,Title1_Prefix=
,Title1_Suffix=
,Title4= /* Title4 is optional */
,Title4_Color=Black
,TitlesJustify=Left /* Center is the SAS default.
                    Unlike web graphs to HTML,
                    for which titles can be pushed out to the HTML source
                    by using ODS HTML ... NOGTITLE,
                    for graphs to disk
                    the titles must be imbedded in the image file.
                    However, until it is changed,
                    SGPLOT centering of titles for HBAR charts is defective:
                    i.e., titles short enough to fit are centered over the bars,
                    while titles that are too long
                    are centered within the full width of the graph.
                    One would expect center justified titles
                    to always be centered within the full width of the graph.
                    For now, Left Justification is the better choice. */
,GraphFileName= /* If GraphFileName is not assigned,
                  then the graph title is compressed and used as a default,
                  but that might have adverse consequences. */
,BorderOnOrOff=OFF /* OFF = my preferred choice for PowerPoint;
                   ON = my preferred choice for Word.
                   After manual Insert Picture in Word,
                   I also turn on the Word border */
,TopN=
,BarLabelVar=
,BarMeasureVar=
,BarMeasureFormat=
,BarMeasureDescription= /* Macro does not retrieve SAS var label even if one exists.
                        If BarMeasureDescription not specified,
                        then SAS var name (BarMeasureVar) is used instead. */
,BarColor=CXCCCCC /* light grey */
,BarWidth=1 /* adjust to less than 1 if needed to reduce barwidth to bar label height
            which is controlled by TextFontPointSize */
,Yoffsetmax=
,TitleFont='Albany AMT/Bold'
,TitleFontPointSize=16 /* adjust to avoid TITLE line wrap,
                       or to reduce vertical space consumption
                       in order to help avoid thinning of bar labels
                       (no WARNING in SAS log when either phenomenon occurs) */
,TextFont='Albany AMT/Bold'
,TextFontPointSize=14 /* adjust to avoid thinning of bar labels
                      (no WARNING in SAS log when it occurs) */
,Xpixels=
,Ypixels= /* adjust to avoid thinning of bar labels
          (no WARNING in SAS log when it occurs) */
);

%macro RemoveCraphAreaFrameFromStyle
(ParentStyle=,StyleWithNoFrame=);

PROC TEMPLATE;
DEFINE STYLE &StyleWithNoFrame;
  PARENT=&ParentStyle;
  CLASS GRAPHWALLS / FRAMEBORDER=OFF; /* remove a useless box around the bars */
```



```

END; RUN;
%mend RemoveCraphAreaFrameFromStyle;

%RemoveCraphAreaFrameFromStyle
(ParentStyle=Styles.Minimal
,StyleWithNoFrame=Styles.MinimalWithNoFrame);

%if %length(&BarMeasureDescription) EQ 0
%then %let BarMeasureDescription = &BarMeasureVar;

DATA _NULL_;
RunDayDateTimeText = 'Run on ' ||
                     TRIM(LEFT(PUT(DATE(),weekdatx37.))) ||
                     ' at ' ||
                     TRIM(LEFT(PUT(TIME(),timeampm11.)));
CALL SYMPUT('RunDayDateTime',TRIM(LEFT(RunDayDateTimeText)));
RUN;

PROC SUMMARY DATA=&data nway;
CLASS &BarLabelVar;
VAR &BarMeasureVar;
OUTPUT OUT=Summed(drop=_type_ _freq_) SUM=;
RUN;

PROC SQL NOPRINT;
SELECT COUNT(&BarLabelVar) , SUM(&BarMeasureVar)
      INTO :CountOfAllBars      , :BarMeasureGrandTotal
      FROM Summed;
QUIT;

PROC SORT DATA=Summed;
BY DESCENDING &BarMeasureVar;
RUN;

DATA All_WithBarLabels(drop=BarLabelSuffix);
LENGTH BarLabel $ 256 BarLabelSuffix $ 7;
SET Summed;
BarLabelSuffix =
  TRIM(LEFT(PUT(((&BarMeasureVar / &BarMeasureGrandTotal) * 100),6.3))) || '%';
IF LENGTH(BarLabelSuffix) EQ 6
THEN BarLabelSuffix = '0' || BarLabelSuffix;
BarLabel = TRIM(LEFT(PUT(_N_,z3.))) /* a bar count more than three digits
                                   will not fit on a PowerPoint slide or Word page */
          || ' : ' || TRIM(LEFT(&BarLabelVar))
          || ' : ' || TRIM(LEFT(PUT(&BarMeasureVar,&BarMeasureFormat)))
          || ' : ' || BarLabelSuffix;
RUN;

OPTIONS OBS=&TopN;

DATA Selected_WithBarLabels;
SET All_WithBarLabels;
RUN;

OPTIONS OBS=MAX;

DATA ToChart;
SET Selected_WithBarLabels;
RUN;

PROC SQL NOPRINT;
SELECT COUNT(&BarLabelVar) , SUM(&BarMeasureVar)
      INTO :CountOfSelectedBars , :BarMeasureSubTotal

```

```

FROM ToChart;
QUIT;

DATA _NULL_;
LENGTH ForSYMPUT ForSYMPUT1 ForSYMPUT2 8;
ForSYMPUT = &BarMeasureGrandTotal;
CALL SYMPUT('GrandTotal' ,TRIM(LEFT(PUT(ForSYMPUT,&BarMeasureFormat))));
ForSYMPUT = &BarMeasureSubTotal;
CALL SYMPUT('SubTotal' ,TRIM(LEFT(PUT(ForSYMPUT,&BarMeasureFormat))));
ForSYMPUT = &CountOfAllBars;
CALL SYMPUT('CountOfAll' ,TRIM(LEFT(PUT(ForSYMPUT,comma32.))));
ForSYMPUT = &CountOfSelectedBars;
CALL SYMPUT('CountOfSelected',TRIM(LEFT(PUT(ForSYMPUT,3.))));
/* Word page or PowerPoint slide will never fit more than a 3-digit bar count */
ForSYMPUT1 = &BarMeasureSubTotal;
ForSYMPUT2 = &BarMeasureGrandTotal;
CALL SYMPUT('SubTotalPercentOfGrandTotal',
  TRIM(LEFT(PUT(( ForSYMPUT1 / ForSYMPUT2 ) * 100 , 6.3))));
RUN;

%if %eval(&TopN GE &CountOfAllBars)
%then %do;
  %let Title1 = &Title1_Prefix.%str( All )&CountOfSelected &Title1_Suffix;
  %let Title2 = All &CountOfSelected &Title1_Suffix Have;
  %let Title3 = Grand Total &BarMeasureDescription &GrandTotal;
%end;
%else %do;
  %let Title1 = &Title1_Prefix.%str( Top )&CountOfSelected &Title1_Suffix;
  %let Title2 = Top &CountOfSelected &Title1_Suffix
    Account for SubTotal &BarMeasureDescription &SubTotal
    which is &SubTotalPercentOfGrandTotal% of Grand Total;
  %let Title2 = %sysfunc(compbl(%nrquote(&Title2)));
  %let Title3 =
    All &CountOfAll &Title1_Suffix Have Grand Total &BarMeasureDescription
&GrandTotal;
%end;

%if %length(&GraphFileName) EQ 0
%then %let GraphFileName = %sysfunc(COMPRESS(&Title1,' '));

ODS GRAPHICS ON / RESET=ALL BORDER=&BorderOnOrOff SCALE=OFF
  HEIGHT=&Ypixels.px WIDTH=&Xpixels.px
  IMAGENAME="&GraphFileName";

FOOTNOTE;

ODS NORESULTS;
ODS LISTING GPATH="&OutputDiskPath" STYLE=Styles.MinimalWithNoFrame;

TITLE1 JUSTIFY=&TitlesJustify FONT=&TitleFont HEIGHT=&TitleFontPointSize PT "&Title1";
TITLE2 JUSTIFY=&TitlesJustify FONT=&TitleFont HEIGHT=&TitleFontPointSize PT "&Title2";
TITLE3 JUSTIFY=&TitlesJustify FONT=&TitleFont HEIGHT=&TitleFontPointSize PT "&Title3";
%if %length(&Title4) NE 0 %then %do;
TITLE4 JUSTIFY=&TitlesJustify FONT=&TitleFont HEIGHT=&TitleFontPointSize PT
  COLOR=&Title4_Color "&Title4";
%end;
%if %upcase(&ShowRunDayDateTime) NE 0 %then %do;
  %if %length(&Title4) EQ 0
  %then %do;
TITLE4
  %end;
  %else %do;
TITLE5

```

```

    %end;
        JUSTIFY=&TitlesJustify FONT=&TitleFont HEIGHT=&TitleFontPointSize PT
"&RunDayDateTime";
    %end;

PROC SGPLOT DATA=ToChart;
HBAR BarLabel / RESPONSE=&BarMeasureVar
    CATEGORYORDER=RESPDESC BARWIDTH=&BarWidth FILL FILLATTRS=(COLOR=&BarColor) OUTLINE;
YAXIS DISPLAY=(NOLABEL NOLINE NOTICKS) VALUEATTRS=(SIZE=&TextFontPointSize)
%if %length(&Yoffsetmax) NE 0 %then %do;
    OFFSETMAX=&Yoffsetmax
%end;
;
XAXIS DISPLAY=NONE;
RUN;

ODS LISTING Style=Styles.Listing;

%mend SubsettedRankingHbarChartsToDisk;

```

NOTE: The macro above is a bit simpler than the prior one that produces two interlinked web graphs. Nevertheless, it, too, might look rather daunting. If you like what it does As Is, just file it in a macro library, use `OPTION SASAUTOS=` to point your SAS program at the macro library, and invoke the macro in a manner shown in the sample invocations. Of course, you also can modify the macro if you wish.

TOOL 5. THE UpTo4WaysRankingHbarChartsToHTML MACRO: UP TO FOUR WAYS TO DO THE SUBSETTED AND RANKED HORIZONTAL BAR CHART (AND DELIVER SUPPORTING DETAIL IF ANY) WITH JUST ONE MACRO INVOCATION, AND BE ABLE TO SUPPORT MULTIPLE SCREEN RESOLUTIONS FOR THE DISPLAY

Well, after completing the macros above, I could not let go.

My newest macro has several added features.

It creates up to four interlinked web graphs, depending on how you invoke it. You can specify any of the following options alone or in any combination:

TopN – for the Top N categories

TopPercent – for enough of the Top categories to account for at least the specified percent of the grand total of the measure of interest

Min – all of the categories for which the measure of interest is greater than or equal to the specified minimum

ShowAll – list all of the categories in rank order

It, optionally, provides hyperlinked spreadsheets: (a) a link is added on the web page to a table of the final input used to create the chart (and that spreadsheet contains a link back to the web page); and (b) each bar is linked to a detail table for all of the variables in the data set and those rows that were summarized to provide input for that bar on the chart. If the data set was only one row per category value, then the raw detail tables for the bars are one-row tables.

ALT text, to present the bar label information (which contains rank, description, value, and percent of whole) in a pop-up box when the mouse is rested on any bar, is turned on if spreadsheets are requested, or if the web graphs are being created on a high resolution screen. On a high resolution screen, more bars can be accommodated, but, for large enough bar counts, a very small font size is used for the bar labels and they might be hard to read.

It sizes the web graphs (except for the ShowAll chart) so that they fit in the browser window and need no scrolling.

The images can be sized for either of two alternatives:

the current frequent maximum resolution of 1920 X 1080 (wide screen laptop monitors, designed, I think, for viewing movies);

and

the most common older resolution of 1024 X 768 (suited to old style monitors with screen width-to-height proportions of 4 to 3, which still are the proportions of most conference room projection screens).

Though the Top N concept of ranking has a long tradition, the more rational ways to subset categories, in my judgement, are with the Top Percent or a minimum threshold.

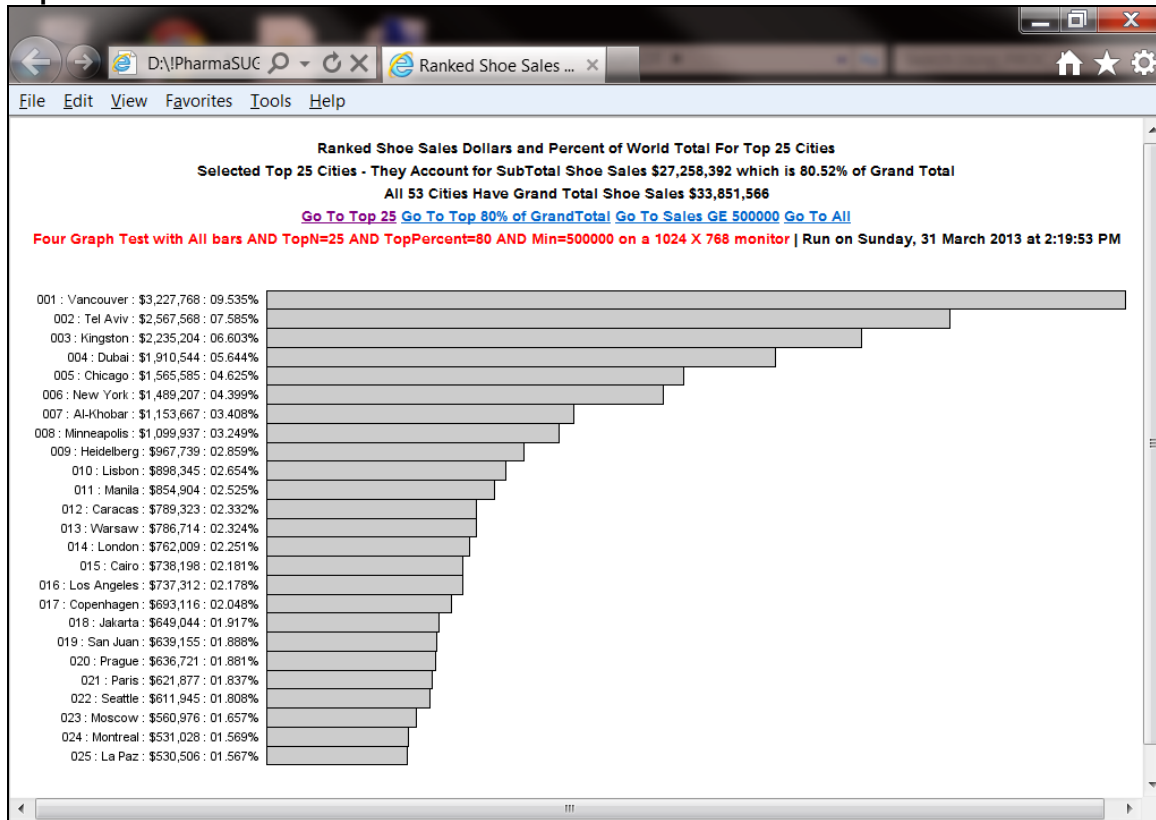
Being able to bring focus to “Show Them What’s Important” in three different ways, plus providing the option to see the whole picture, are a maximally effective way to visually communicate the significance of the data.

The logic in the macro to support the two different resolutions mentioned above can readily be adapted to support other resolutions as well. It takes some effort and rounds of testing, but the process is straightforward. If interested in an explanation, please notify me of your need via email.

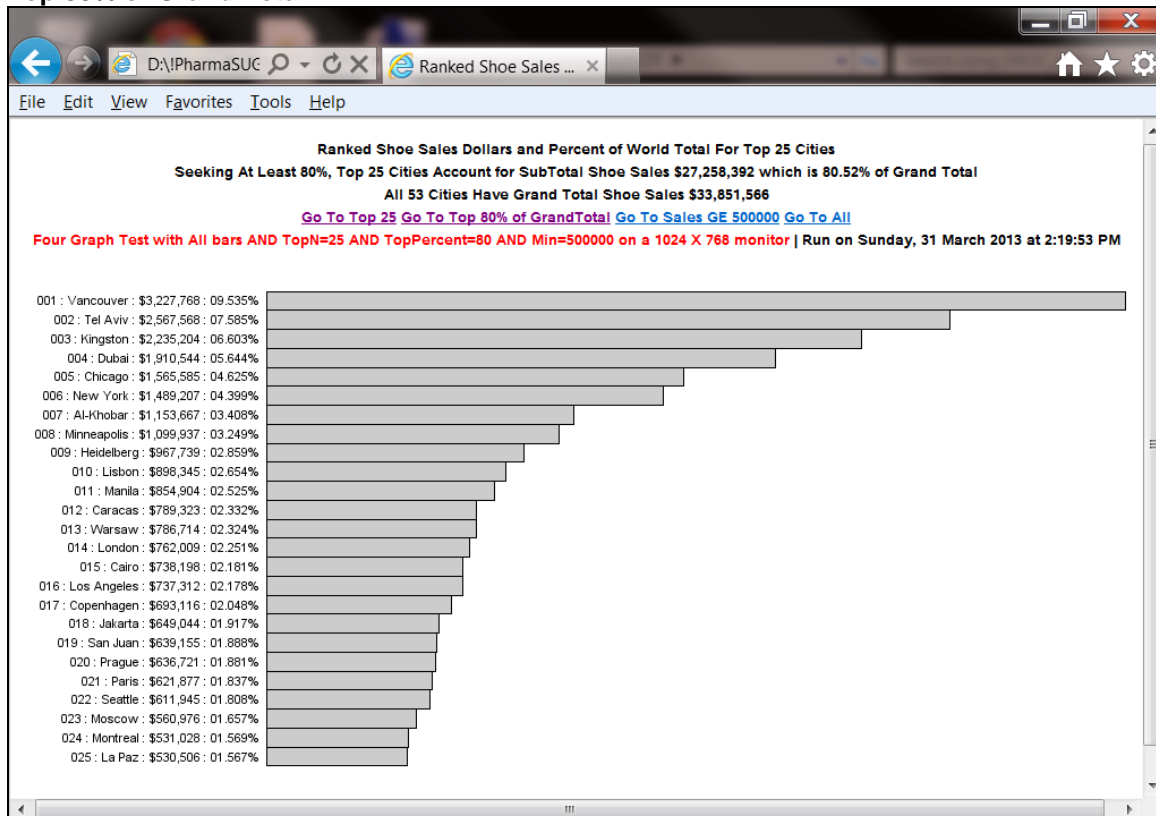
On the following pages are screen images from various scenarios, followed by code for the macro invocations used and then the full macro itself.

NOTE: This macro requires SAS Version 9.3 or higher, and relies on use of SAS ODS Graphics.

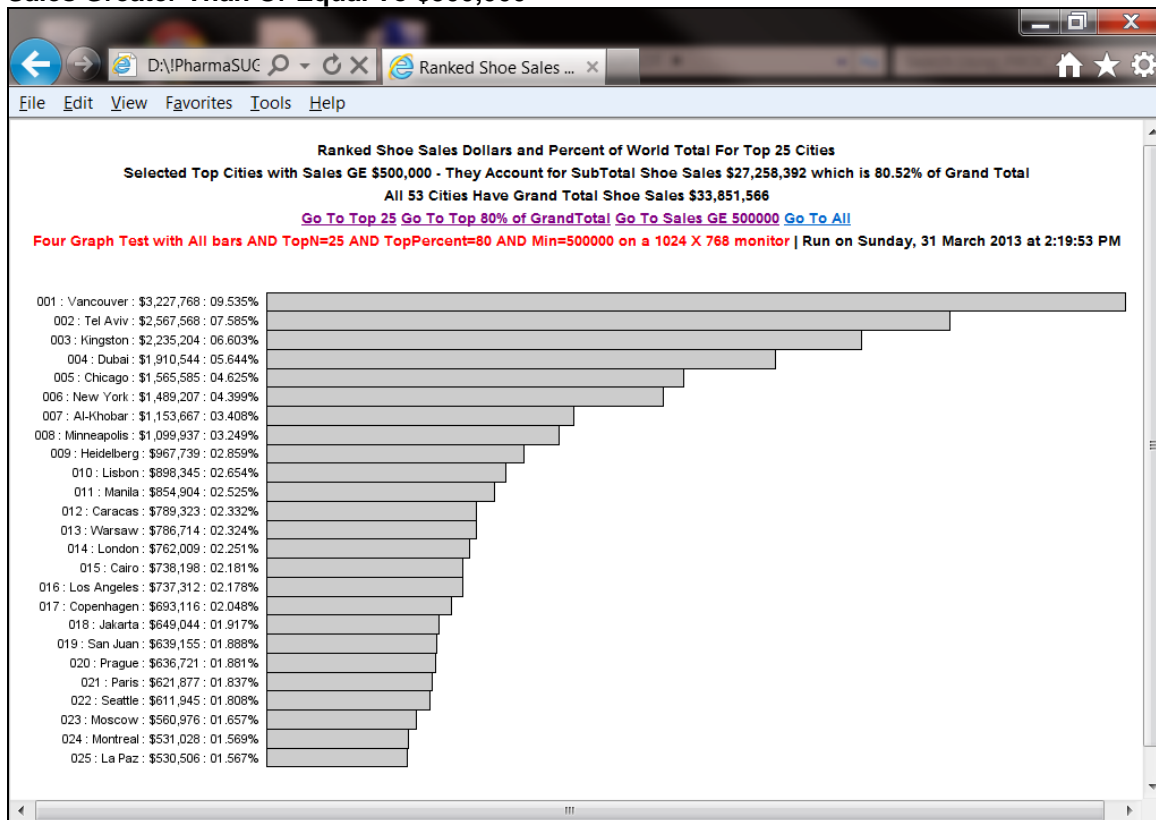
Top 25



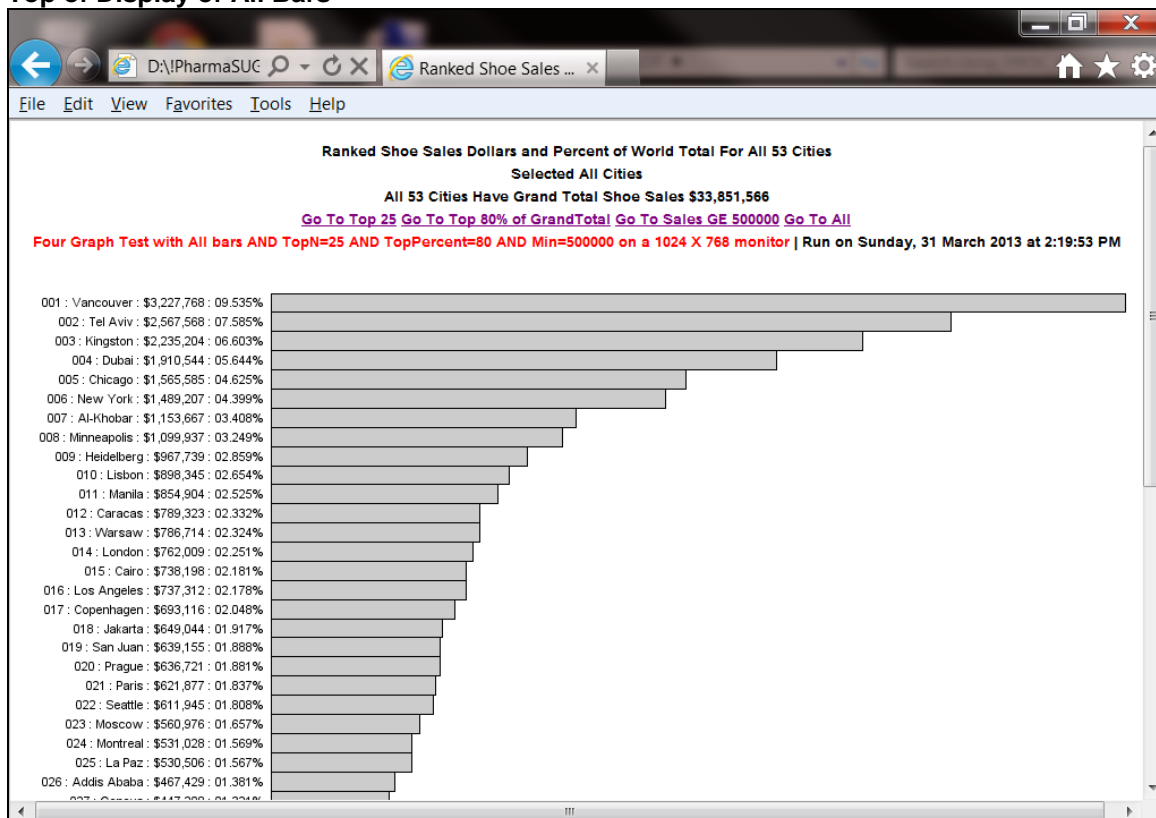
Top 80% of Grand Total



Sales Greater Than Or Equal To \$500,000



Top of Display of All Bars



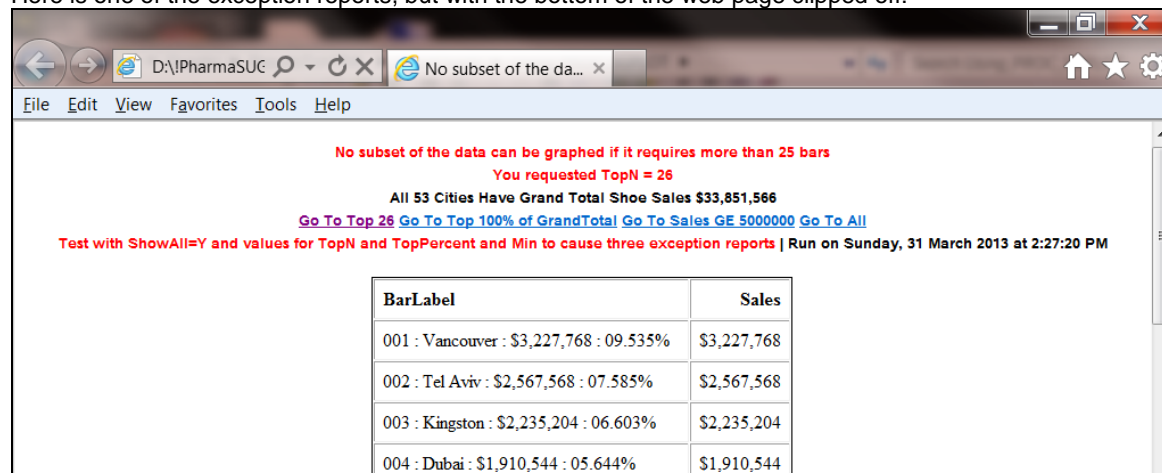
Below is the macro invocation code used to create the four preceding charts. The values of TopN, TopPercent, and Min were chosen to yield the same number of bars.

```
%UpTo4WaysRankingHbarChartsToHTML(
data=SASHELP.SHOES
,For1024X768monitorsInstead=YES
,ShowAll=YES
,TopN=25
,TopPercent=80
,Min=500000
,RptPath=D:\PharmaSUG2013\Graphs\ResultsToHTML\HbarCharts\Using_PROC_SGPLOT
,Title1_Prefix=Ranked Shoe Sales Dollars and Percent of World Total For
,Title1_Suffix=Cities
,Title5=Four Graph Test with All bars AND TopN=25 AND TopPercent=80 AND Min=500000 on
a 1024 X 768 monitor /* Title5 is optional */
,Title5_Color=red /* black is the default */
,RunDayDateTimeInTitle5=YES /* YES is the default. You can override with NO. */
,RunDayDateTimeAsFileNameSuffix=YES /* NO is the default */
,BarLabelVar=Subsidiary
,BarMeasureVar=Sales
,BarMeasureDescription=Shoe Sales
,BarMeasureFormat=dollar11.);
```

Here is macro invocation code used to create three exception reports and an All Bars chart.

```
%UpTo4WaysRankingHbarChartsToHTML(
data=SASHELP.SHOES
,For1024X768monitorsInstead=YES
,ShowAll=YES
,TopN=26 /* cause an exception report for TopN Subset Chart */
,TopPercent=100 /* cause an exception report for TopPercent Subset Chart */
,Min=5000000 /* cause an exception report for Min Subset Chart */
,RptPath=D:\PharmaSUG2013\Graphs\ResultsToHTML\HbarCharts\Using_PROC_SGPLOT
,Title1_Prefix=Ranked Shoe Sales Dollars and Percent of World Total For
,Title1_Suffix=Cities
,Title5=Test with ShowAll=Y and values for TopN and TopPercent and Min to cause three
exception reports /* Title5 is optional */
,Title5_Color=red /* black is the default */
,RunDayDateTimeInTitle5=YES /* YES is the default. You can override with NO. */
,RunDayDateTimeAsFileNameSuffix=YES /* NO is the default */
,BarLabelVar=Subsidiary
,BarMeasureVar=Sales
,BarMeasureDescription=Shoe Sales
,BarMeasureFormat=dollar11.);
```

Here is one of the exception reports, but with the bottom of the web page clipped off:



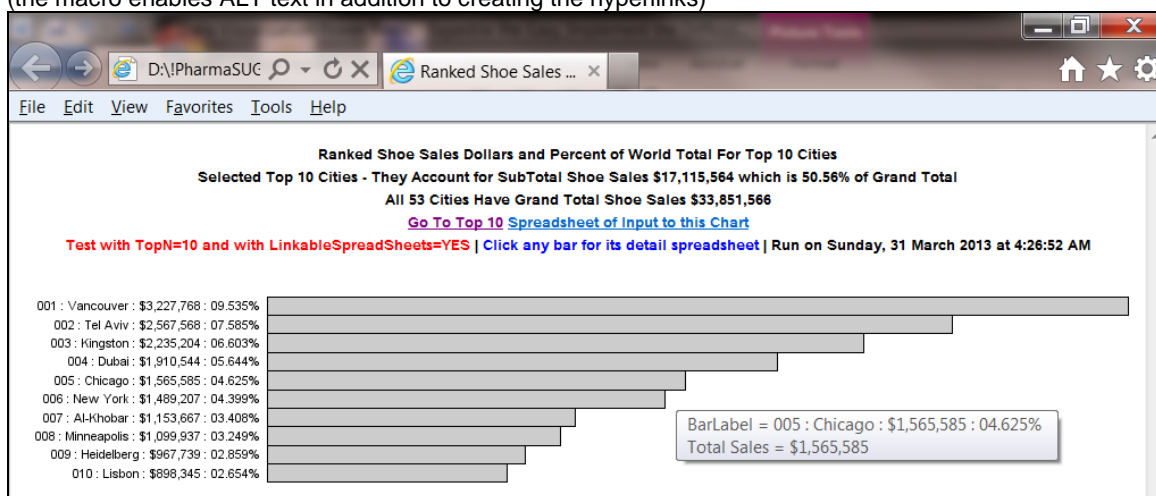
No subset of the data can be graphed if it requires more than 25 bars
You requested TopN = 26
All 53 Cities Have Grand Total Shoe Sales \$33,851,566
[Go To Top 26](#) [Go To Top 100% of GrandTotal](#) [Go To Sales GE 5000000](#) [Go To All](#)
Test with ShowAll=Y and values for TopN and TopPercent and Min to cause three exception reports | Run on Sunday, 31 March 2013 at 2:27:20 PM

BarLabel	Sales
001 : Vancouver : \$3,227,768 : 09.535%	\$3,227,768
002 : Tel Aviv : \$2,567,568 : 07.585%	\$2,567,568
003 : Kingston : \$2,235,204 : 06.603%	\$2,235,204
004 : Dubai : \$1,910,544 : 05.644%	\$1,910,544

Here is code to create a Top 10 chart in which every bar is hyperlinked to a spreadsheet of its supporting detail, which was summarized, and in which the bar chart web page is hyperlinked to the spreadsheet of its input (that spreadsheet, shown on the next page, is hyperlinked back to the bar chart):

```
%UpTo4WaysRankingHbarChartsToHTML(
data=SASHELP.SHOES
,For1024X768monitorsInstead=YES
,LinkableSpreadSheets=YES /* NO is the default */
,TopN=10
,RptPath=D:\PharmaSUG2013\Graphs\ResultsToHTML\HbarCharts\Using_PROC_SGPLOT
,Title1_Prefix=Ranked Shoe Sales Dollars and Percent of World Total For
,Title1_Suffix=Cities
,Title5=Test with TopN=10 and with LinkableSpreadSheets=YES /* Title5 is optional */
,Title5_Color=red /* black is the default */
,RunDayDateInTitle5=YES /* YES is the default. You can override with NO. */
,RunDayDateTimeAsFileNameSuffix=YES /* NO is the default */
,BarLabelVar=Subsidiary
,BarMeasureVar=Sales
,BarMeasureDescription=Shoe Sales
,BarMeasureFormat=dollar11.);
```

Top 10 chart, with mouse about to be clicked on the Chicago bar, but with bottom of the web page clipped off (the macro enables ALT text in addition to creating the hyperlinks)



After clicking on the Chicago bar,
 Spreadsheet of detail for Chicago, but with the bottom of the Excel window clipped off

President of the United States, but with the bottom of the Excel window clipped off.

Chicago_Sun_31MAR2013_042652_594.xls - Microsoft Excel

IDX	Subsidiary							H	I	J	K
	A	B	C	D	E	F	G				
1	Subsidiary	Region	Product	Stores	Sales	Inventory	Returns				
2	Chicago	United States	Men's Casual	26	\$408,978	\$831,672	\$17,907				
3	Chicago	United States	Slipper	13	\$329,235	\$1,049,530	\$10,654				
4	Chicago	United States	Women's Dress	37	\$293,313	\$754,157	\$10,194				
5	Chicago	United States	Men's Dress	28	\$261,607	\$824,485	\$8,915				
6	Chicago	United States	Women's Casual	11	\$172,021	\$474,524	\$5,399				
7	Chicago	United States	Boot	16	\$82,483	\$305,061	\$3,735				
8	Chicago	United States	Sport Shoe	17	\$17,347	\$74,605	\$684				
9	Chicago	United States	Sandal	2	\$601	\$17,121	\$35				
10					\$1,565,585						
11											

After clicking on the hyperlink in the subtitle of the web bar chart, here is the spreadsheet of its input data:

Microsoft Excel window showing the spreadsheet 'RankedShoeSalesDollarsandPercentofWorldTotalForCities_TOPN_Sun_31MAR2013_042652_594.XLS'.

Subsidiary	BarLabel	Total Sales
Vancouver	001 : Vancouver : \$3,227,768 : 09.535%	\$3,227,768
Tel Aviv	002 : Tel Aviv : \$2,567,568 : 07.585%	\$2,567,568
Kingston	003 : Kingston : \$2,235,204 : 06.603%	\$2,235,204
Dubai	004 : Dubai : \$1,910,544 : 05.644%	\$1,910,544
Chicago	005 : Chicago : \$1,565,585 : 04.625%	\$1,565,585
New York	006 : New York : \$1,489,207 : 04.399%	\$1,489,207
Al-Khobar	007 : Al-Khobar : \$1,153,667 : 03.408%	\$1,153,667
Minneapolis	008 : Minneapolis : \$1,099,937 : 03.249%	\$1,099,937
Heidelberg	009 : Heidelberg : \$967,739 : 02.859%	\$967,739
Lisbon	010 : Lisbon : \$898,345 : 02.654%	\$898,345
		\$17,115,564

In the spreadsheet above, note the hyperlink back to the bar chart.

If you were to decide to adapt the macro to support a resolution other than 1920 X 1080 or 1024 X 768, the development process would entail experimenting with parameters, some of which are macro invocation parameters, and others which would be imbedded inside the macro to customize the revised algorithm required for the alternative new resolution. To make it easier to tie test results to code choices, the macro provides an option to document the state of various parameters.

Here is an example (with the web page bottom clipped off), for the already supported case of 1024 X 768 resolution:



Here is magnification of the diagnostic message:

8 bars, 17 Ypixels per bar, 26 Ypixels overhead, 162 Total Ypixels, 990 Xpixels, Titles 10pt 'Albany AMT/Bold', Bar Labels 8pt 'Albany AMT/Bold'

Here is the macro invocation code, with two key parameter assignments highlighted in blue:

```
%UpTo4WaysRankingHbarChartsToHTML(  
data=SASHELP.SHOES  
,For1024X768monitorsInstead=YES  
,Min=1000000  
,RptPath=D:\!PharmaSUG2013\Graphs\ResultsToHTML\HbarCharts\Using_PROC_SGPLOT  
,Title1_Prefix=Ranked Shoe Sales Dollars and Percent of World Total For  
,Title1_Suffix=Cities  
,UseTitle5ForDiagnostics=YES  
,RunDayDateTimeInTitle5=NO /* Do not show Run Day Date Time in Title5 */  
,RunDayDateTimeAsFileNameSuffix=YES /* NO is the default */  
,BarLabelVar=Subsidiary  
,BarMeasureVar=Sales  
,BarMeasureDescription=Shoe Sales  
,BarMeasureFormat=dollar11.);
```

It was not mandatory to turn off the Run Day Date Time that could appear in Title5.

Since this macro is expected to be dual-use: either for 1024 X 768 resolution or 1920 X 1080 resolution, on the following pages are samples from high-resolution mode. The four-graph set shown was deliberately structured, unlike the low-resolution four-graph set, to deliver significantly different numbers of bars for the three subset cases. If you inspect the charts you will note that the bar widths are the same. When the macro is invoked to present at least one subset, it is the bar width needed for the subset with the largest number of bars that is selected to be used for ALL of the graphs created by that invocation. The bar width is set as wide as possible to still fit within the browser window without scrolling. The font height is set proportionately and as large as possible without causing the ODS graphics machinery to do its unannounced and always unwelcome and anti-communicative action of “thinning” the bar labels and rendering the chart useless. That bar width and that font height are also used for the All Bars chart, if that chart is also selected during the macro invocation.

On the following pages the bar labels are difficult or impossible to read. The screen image is taken from a 17-inch diagonal screen laptop where the image display width is over 14.5 inches and is compressed to fit this page width of 6.5 inches.

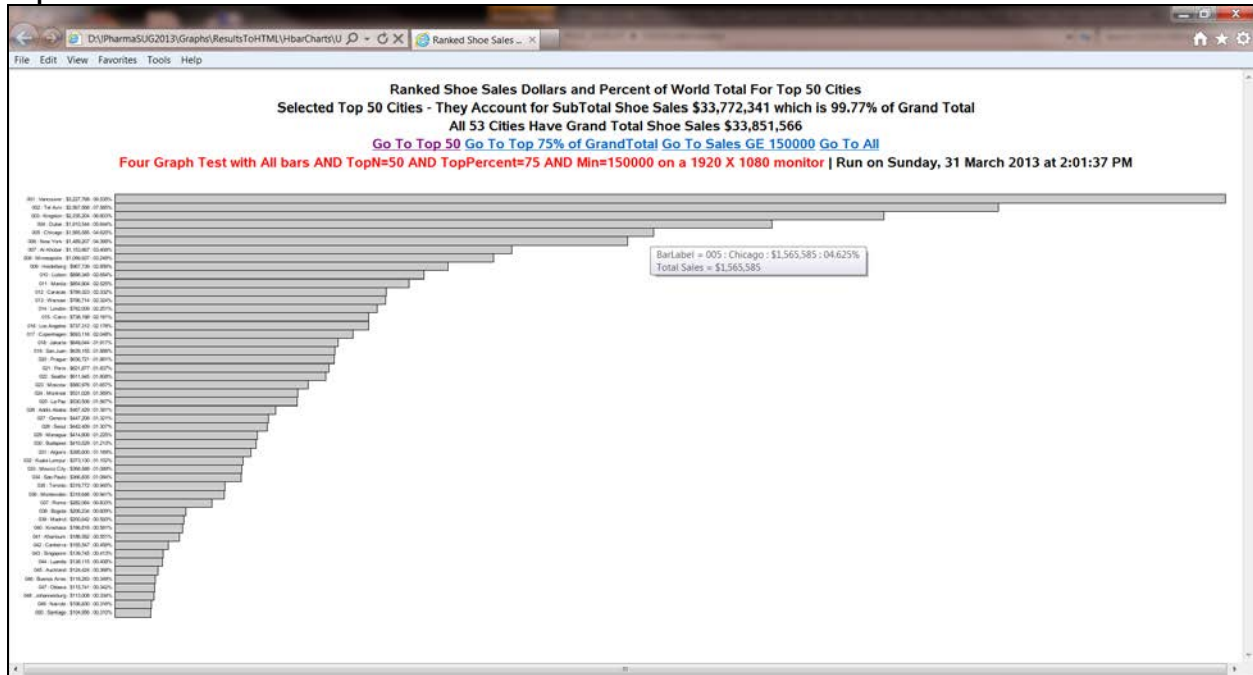
NOTE: When the macro is not used for 1024 X 768 resolution, ALT text is turned on so that the bar label and the response value are displayed as pop-up text. The presumption is that the resolution will be higher and more bars will be displayable, so that the font used for the bar label itself will need to be smaller and possibly harder to read.

Here is the macro invocation code used to create the charts on the following pages:

```
%UpTo4WaysRankingHbarChartsToHTML(  
data=SASHELP.SHOES  
,For1024X768monitorsInstead=NO  
,ShowAll=YES  
,TopN=50  
,TopPercent=75  
,Min=150000  
,RptPath=D:\!PharmaSUG2013\Graphs\ResultsToHTML\HbarCharts\Using_PROC_SGPLOT  
,Title1_Prefix=Ranked Shoe Sales Dollars and Percent of World Total For  
,Title1_Suffix=Cities  
,Title5=Four Graph Test with All bars AND TopN=50 AND TopPercent=75 AND Min=150000 on  
a 1920 X 1080 monitor /* Title5 is optional */  
,Title5_Color=red /* black is the default */  
,RunDayDateTimeInTitle5=YES /* YES is the default. You can override with NO. */  
,RunDayDateTimeAsFileNameSuffix=YES /* NO is the default */  
,BarLabelVar=Subsidiary  
,BarMeasureVar=Sales  
,BarMeasureDescription=Shoe Sales  
,BarMeasureFormat=dollar11.);
```

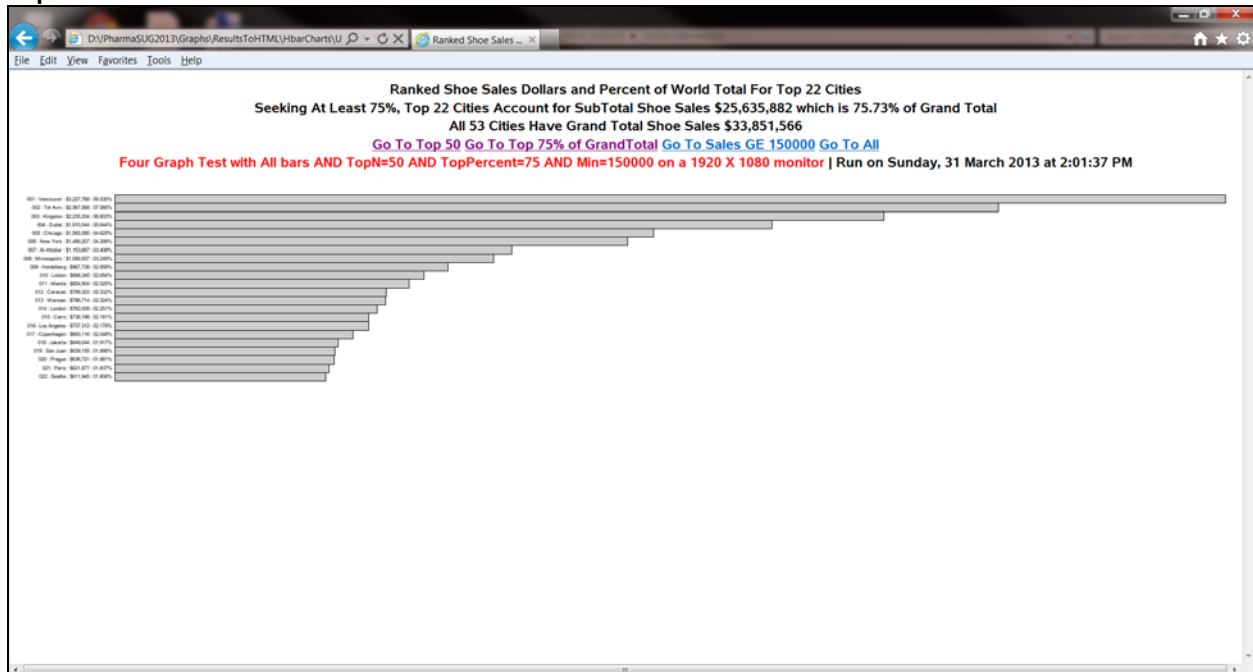
NOTE: In the above and other macro invocations, Title5 is used to describe how the macro was invoked. The macro COULD BE modified to extract that information from its invocation parameters and build that Title5 text as a default.

Top 50



NOTE: On charts for a high-resolution monitor, which accommodates more bars, ALT text is always turned on by the macro automatically, in case the viewer might find the bar labels hard to read.

Top 75% of Grand Total



Ranked Shoe Sales Dollars and Percent of World Total For Top 42 Cities

Selected Top Cities with Sales GE \$150,000 - They Account for SubTotal Shoe Sales \$32,811,239 which is 96.93% of Grand Total

All 53 Cities Have Grand Total Shoe Sales \$33,851,566

[Go To Top 50](#) [Go To Top 75% of GrandTotal](#) [Go To Sales GE 150000](#) [Go To All](#)

Four Graph Test with All bars AND TopN=50 AND TopPercent=75 AND Min=150000 on a 1920 X 1080 monitor | Run on Sunday, 31 March 2013 at 2:01:37 PM

City	Sales	Percent
001 New York	\$2,327,769	6.85%
002 Los Angeles	\$2,167,456	6.39%
003 Chicago	\$2,104,244	6.22%
004 Dallas	\$2,103,944	6.22%
005 New York	\$1,985,557	5.86%
006 New York	\$1,985,557	5.86%
007 New York	\$1,985,557	5.86%
008 New York	\$1,985,557	5.86%
009 New York	\$1,985,557	5.86%
010 New York	\$1,985,557	5.86%
011 New York	\$1,985,557	5.86%
012 New York	\$1,985,557	5.86%
013 New York	\$1,985,557	5.86%
014 New York	\$1,985,557	5.86%
015 New York	\$1,985,557	5.86%
016 New York	\$1,985,557	5.86%
017 New York	\$1,985,557	5.86%
018 New York	\$1,985,557	5.86%
019 New York	\$1,985,557	5.86%
020 New York	\$1,985,557	5.86%
021 New York	\$1,985,557	5.86%
022 New York	\$1,985,557	5.86%
023 New York	\$1,985,557	5.86%
024 New York	\$1,985,557	5.86%
025 New York	\$1,985,557	5.86%
026 New York	\$1,985,557	5.86%
027 New York	\$1,985,557	5.86%
028 New York	\$1,985,557	5.86%
029 New York	\$1,985,557	5.86%
030 New York	\$1,985,557	5.86%
031 New York	\$1,985,557	5.86%
032 New York	\$1,985,557	5.86%
033 New York	\$1,985,557	5.86%
034 New York	\$1,985,557	5.86%
035 New York	\$1,985,557	5.86%
036 New York	\$1,985,557	5.86%
037 New York	\$1,985,557	5.86%
038 New York	\$1,985,557	5.86%
039 New York	\$1,985,557	5.86%
040 New York	\$1,985,557	5.86%
041 New York	\$1,985,557	5.86%
042 New York	\$1,985,557	5.86%

Ranked Shoe Sales Dollars and Percent of World Total For All 53 Cities

Selected All Cities

All 53 Cities Have Grand Total Shoe Sales \$33,851,566

Go To Top 50 Go To Top 75% of GrandTotal Go To Sales GE 150000 Go To All

Four Graph Test with All bars AND TopN=50 AND TopPercent=75 AND Min=150000 on a 1920 X 1080 monitor | Run on Sunday, 31 March 2013 at 2:01:37 PM

City	Sales (\$)	Percent (%)
101. Houston	\$1,037,768	3.03%
102. New York	\$1,037,768	3.03%
103. Houston	\$1,037,768	3.03%
104. Houston	\$1,037,768	3.03%
105. Chicago	\$1,037,768	3.03%
106. New York	\$1,037,768	3.03%
107. Houston	\$1,037,768	3.03%
108. Houston	\$1,037,768	3.03%
109. Houston	\$1,037,768	3.03%
110. Houston	\$1,037,768	3.03%
111. Houston	\$1,037,768	3.03%
112. Houston	\$1,037,768	3.03%
113. Houston	\$1,037,768	3.03%
114. Houston	\$1,037,768	3.03%
115. Houston	\$1,037,768	3.03%
116. Houston	\$1,037,768	3.03%
117. Houston	\$1,037,768	3.03%
118. Houston	\$1,037,768	3.03%
119. Houston	\$1,037,768	3.03%
120. Houston	\$1,037,768	3.03%
121. Houston	\$1,037,768	3.03%
122. Houston	\$1,037,768	3.03%
123. Houston	\$1,037,768	3.03%
124. Houston	\$1,037,768	3.03%
125. Houston	\$1,037,768	3.03%
126. Houston	\$1,037,768	3.03%
127. Houston	\$1,037,768	3.03%
128. Houston	\$1,037,768	3.03%
129. Houston	\$1,037,768	3.03%
130. Houston	\$1,037,768	3.03%
131. Houston	\$1,037,768	3.03%
132. Houston	\$1,037,768	3.03%
133. Houston	\$1,037,768	3.03%
134. Houston	\$1,037,768	3.03%
135. Houston	\$1,037,768	3.03%
136. Houston	\$1,037,768	3.03%
137. Houston	\$1,037,768	3.03%
138. Houston	\$1,037,768	3.03%
139. Houston	\$1,037,768	3.03%
140. Houston	\$1,037,768	3.03%
141. Houston	\$1,037,768	3.03%
142. Houston	\$1,037,768	3.03%
143. Houston	\$1,037,768	3.03%
144. Houston	\$1,037,768	3.03%
145. Houston	\$1,037,768	3.03%
146. Houston	\$1,037,768	3.03%
147. Houston	\$1,037,768	3.03%
148. Houston	\$1,037,768	3.03%
149. Houston	\$1,037,768	3.03%
150. Houston	\$1,037,768	3.03%

Code for the UpTo4WaysRankingHbarChartsToHTML macro is on the following pages. **It could be modified to turn off the hyperlink (on the fourth title line) that is for whatever happens to be the currently displayed web page.**

```

%macro UpTo4WaysRankingHbarChartsToHTML(
data=
,For1024X768monitorsInstead=NO
,MaximumBarCount=50
,LinkableSpreadsheets=NO
,RptPath=
,Title1_Prefix=
,Title1_Suffix=
,BorderOnOrOff=OFF /* Turn this on when making changes to code, in order to
                        see where the edge of the image is in the browser window
                        when creating charts for and viewing them
                        on a monitor with a previously unsupported resolution.
                        Size the image so that no horizontal scrolling is ever required.
                        Ordinarily, the image should be displayed in the browser with no
                        border. The border serves no visual communication purpose. */
,UseTitle5ForDiagnostics=NO /* YES could be useful during testing a revised macro */
,Title5= /* Title5 is optional */
,Title5_Color=black
,RunDayDateTimeInTitle5=YES /* If using TITLE5 for diagnostic
                                or for a custom message,
                                it might be wise to turn off Run Day, Date, & Time
                                to prevent wrap of a too long title line, which could
                                push the bottom of chart below the browser window. */
,RunDayDateTimeAsFileNameSuffix=NO
,TitlesJustify=Center /* Center is actually the SAS default.
                        If titles were imbedded in the graph,
                        SGPLOT HBAR chart centering of titles would be defective.
                        Left Justification would be the better choice.
                        ODS HTML NOGTITLE (used below in macro) makes Center OK. */
,ShowAll=
,TopN= /* subset number of bars to be shown */
,TopPercent= /* subset enough bars to account for this percent of the grand total */
,Min= /* subset all bars with BarMeasureVar GE this value */
,BarLabelVar=
,BarMeasureVar=
,BarMeasureFormat=
,BarMeasureDescription= /* Macro does not retrieve SAS var label even if one exists.
                        If BarMeasureDescription not specified,
                        then SAS var name (BarMeasureVar) is used instead. */
,BarColor=CXCCCCC /* light grey */
,TitleFont='Albany AMT/Bold'
,TitleFontPointSize=16
,TextFont='Albany AMT/Bold'
,Xpixels=1875 /* 1875 is maximum to avoid horizontal scrolling on 1920 X 1080 monitor.
                Select a value based on the expected viewing monitor.
                REDUCE Xpixels if java heap space out of memory & SGPLOT failure
                when bar count of the All Bars companion chart is high.
                The memory problem is announced with these messages in the SAS log:
                ERROR: Java virtual machine exception.
                        java.lang.OutOfMemoryError: Java heap space.
                NOTE: The SAS System stopped processing this step because of errors.
                Alternatively, you might get relief by changing the SASV9.CFG file,
                following the instructions
                in SAS Note http://support.sas.com/kb/31/184.html */
,YpixelsOverhead=31 /* 31 is the value for a 1920 X 1080 monitor. This extra space in
                        the image is used for nondisplayable purposes, not the bars.
                        31 YpixelsOverhead is for use with TextFont='Albany AMT/Bold'
                        and with the rule inside the macro for choosing
                        TextFontPointSize and YpixelsPerBar based on the
                        maximum number of bars that qualify for any subset(s) requested.
                        YpixelsOverhead, YpixelsPerBar, TopN are used inside the macro
                        to compute the total Y pixels for height of the image file. */
);

```

```

*****;
* Macro: UpTo4WaysRankingHbarChartsToHTML *;
* Filename: UpTo4WaysRankingHbarChartsToHTML.sas *;
* Author: LeRoy Bessler PhD *;
* Email: Le_Roy_Bessler@wi.rr.com *;
* Notes: Not all of the macro invocation parameters are edited for feasibility *;
* Usage Note: any one, two, three, or all four of the parameters *;
*             ShowAll, TopN, TopPercent, Min *;
*             can be used. *;
* Environment: Must be run on SAS Version 9.3 or higher. *;
*             Unless the macro is modified, *;
*             its graphs can be viewed at 1920 X 1080 resolution *;
*             or 1024 X 768 resolution. *;
*             A 1024 X 768 image will be viewable at 1920 X 1080 resolution, *;
*             but a 1920 X 1080 image will require scrolling at 1024 X 768. *;
* Design: For the original version of this macro, *;
*         parameters were optimized for monitor 1024 pixels by 1080 pixels. *;
*         They must be set to suit the target users of your charts. *;
*         The objective is that SUBSET charts should require no scrolling. *;
*         For ShowAll=YES, scrolling is expected when bar count is large. *;
*         The maximum bar count supported for subset charts is 50 or 25. *;
*         Actual bar count using the TopPercent or Min option is unpredictable. *;
*         If too many bars qualify (i.e., more than the maximum *;
*         an exception report is provided in the web page *;
*         with a listing of the data that would have been charted. *;
*         The default text font is shipped with SAS, *;
*         and its use is imbedded in the image files. So, the viewer *;
*         of the web graph does not need the font on the viewing computer. *;
*         If you modify the code to change the ODS style, *;
*         the results are unpredictable by me. *;
*         If the viewer of the web graphs has extra toolbars active *;
*         in the web browser, subset image files might extend off the screen. *;
*         The web graphs were viewed with Internet Explorer 9 *;
*         with only the Menu Bar and Status Bar in use. *;
*****;

%if %upcase(&For1024X768monitorsInstead) EQ YES
%then %do;
    %let Xpixels = 990;          /* 1875 for 1920 X 1080 */
    %let MaximumBarCount = 25;   /* 50   for 1920 X 1080 */
    %let TitleFontPointSize = 10; /* 16   for 1920 X 1080 */
    %let YpixelsOverhead = 26;   /* 31   for 1920 X 1080 */
    %let TextFontPtSizeFor1024X768monitor = 8; /* 16 for 1920 X 1080 */
    %let YpixelsPerBarFor1014X768monitor = 17; /* ranges from 26 to 13
        as subset maximum bar count ranges from 25 to 50 */
%end;

%if %upcase(&UseTitle5ForDiagnostics) EQ YES AND %length(&Title5) NE 0 %then %do;
    %put USER ERROR: With the &sysmacroname macro, if you specify a value for Title5,
you must specify UseTitle5ForDiagnostics=NO;
    %GoTo MacExit;
%end;

%if "%upcase(&TextFont)" NE "'ALBANY AMT/BOLD'" %then %do;
    %put USER ERROR: This &sysmacroname macro is currently configured to only work with
the Albany AMT/Bold as TextFont parameter;
    %GoTo MacExit;
%end;

%if %length(&ShowAll&TopN&TopPercent&Min) EQ 0 %then %do;
    %put USER ERROR: No chart type was selected;
    %GoTo MacExit;

```

```

%end;

%if %length(&TopN) NE 0 %then %do;
  %if %sysfunc(NOTDIGIT(&TopN)) NE 0 %then %do;
    %put USER ERROR: You specified TopN=&TopN but TopN must be numeric;
    %GoTo MacExit;
  %end;
  %if &TopN EQ 0 %then %do;
    %put USER ERROR: You specified TopN=&TopN but TopN must be greater than zero;
    %GoTo MacExit;
  %end;
%end;

%if %length(&TopPercent) NE 0 %then %do;
  %if %sysfunc(NOTDIGIT(&TopPercent)) NE 0 %then %do;
    %put USER ERROR: You specified TopPercent=&TopPercent but TopPercent must be
numeric;
    %GoTo MacExit;
  %end;
  %if &TopPercent EQ 0 %then %do;
    %put USER ERROR: You specified TopPercent=&TopPercent but TopPercent must be
greater than zero;
    %GoTo MacExit;
  %end;
  %if &TopPercent GT 100 %then %do;
    %put USER ERROR: You specified TopPercent=&TopPercent but TopPercent must not
exceed 100;
    %GoTo MacExit;
  %end;
%end;

%if %length(&Min) NE 0 %then %do;
  %if %sysfunc(NOTDIGIT(&Min)) NE 0 %then %do;
    %put USER ERROR: You specified Min=&Min but Min must be numeric;
    %GoTo MacExit;
  %end;
%end;

%if %length(&BarMeasureDescription) EQ 0
%then %let BarMeasureDescription = &BarMeasureVar;

DATA _NULL_;
RunDate = DATE();
RunTime = TIME();
RunDayDateTimeText =
  'Run on ' ||
  TRIM(LEFT(PUT(RunDate,weekdatx37.))) || ' at ' ||
  TRIM(LEFT(PUT(RunTime,timeampm11.)));
CALL SYMPUT('RunDayDateTime',TRIM(LEFT(RunDayDateTimeText)));
RunDayDateTimeFileNameSuffix =
  TRIM(LEFT(PUT(RunDate,downname3.))) || '_' ||
  TRIM(LEFT(PUT(RunDate,date9.))) || '_' ||
  TRIM(LEFT(COMPRESS(PUT(RunTime,TOD8.),':'))) || '_' ||
  SUBSTR(PUT(RunTime,TOD12.3),10,3) /* prevent duplicate timestamps
when two successive macro invocations run during the same second */
;
CALL SYMPUT('FileNameDTsuffix',TRIM(LEFT(RunDayDateTimeFileNameSuffix)));
RUN;

%if %upcase(&RunDayDateTimeAsFileNameSuffix) EQ YES
%then %let FileNameDTsuffix = %str(&_amp;FileNameDTsuffix);
%else %let FileNameDTsuffix = %str();

```

```

PROC SUMMARY DATA=&data nway;
CLASS &BarLabelVar;
VAR &BarMeasureVar;
OUTPUT OUT=Summed(drop=_type_ _freq_) SUM=;
RUN;

PROC SQL NOPRINT;
SELECT COUNT(&BarLabelVar)      , SUM(&BarMeasureVar)
      INTO :CountOfAllBars trimmed , :BarMeasureGrandTotal
      FROM Summed;
QUIT;

DATA _NULL_;
LENGTH ForSYMPUT 8;
ForSYMPUT = &BarMeasureGrandTotal;
CALL SYMPUT('GrandTotal',TRIM(LEFT(PUT(ForSYMPUT,&BarMeasureFormat))));
RUN;

PROC SORT DATA=Summed;
BY DESCENDING &BarMeasureVar;
RUN;

DATA ALLtoChart_WithBarLabels(drop=BarLabelSuffix);
LENGTH BarLabel $ 256 BarLabelSuffix $ 7;
SET Summed;
BarLabelSuffix =
  TRIM(LEFT(PUT(((&BarMeasureVar / &BarMeasureGrandTotal) * 100),6.3))) || '%';
IF LENGTH(BarLabelSuffix) EQ 6
THEN BarLabelSuffix = '0' || BarLabelSuffix;
BarLabel = TRIM(LEFT(PUT(_N_,z3.))) || ' : ' || /* assumes max possible is 999 bars */
  TRIM(LEFT(&BarLabelVar)) || ' : ' ||
  TRIM(LEFT(PUT(&BarMeasureVar,&BarMeasureFormat))) || ' : ' || BarLabelSuffix;
RUN;

%let TopNSubTotalCount      = 0;
%let TopPercentSubTotalCount = 0;
%let GEMinSubTotalCount     = 0;

%let TopNChartPossible      = NO;
%let TopPercentChartPossible = NO;
%let GEMinChartPossible     = NO;

%if %length(&TopN) NE 0 %then %do;

OPTIONS OBS=&TopN;

DATA TOPNtoChart_WithBarLabels;
SET ALLtoChart_WithBarLabels;
RUN;

OPTIONS OBS=MAX;

PROC SQL NOPRINT;
SELECT SUM(&BarMeasureVar) , Count(*)
      INTO :BarMeasureSubTotal , :TopNSubTotalCount trimmed
      FROM TopNtoChart_WithBarLabels;
QUIT;

DATA _NULL_;
LENGTH ForSYMPUT ForSYMPUT1 ForSYMPUT2 8;
ForSYMPUT = &BarMeasureSubTotal;
CALL SYMPUT('TopNSubTotal',TRIM(LEFT(PUT(ForSYMPUT,&BarMeasureFormat))));
ForSYMPUT1 = &BarMeasureSubTotal;

```



```

ForSYMPUT2 = &BarMeasureGrandTotal;
CALL SYMPUT('TopNSubTotalPercent',TRIM(LEFT(PUT(( ForSYMPUT1 / ForSYMPUT2 ) * 100 ,
6.2 ))));
RUN;

%if %eval(&TopNSubTotalCount LE &MaximumBarCount)
%then %let TopNChartPossible = YES;
%else %do;
%let TopN_Title1 = No subset of the data can be graphed if it requires more than
&MaximumBarCount bars;
%let TopN_Title2 = You requested TopN = &TopN;
%let TopNChartPossible = NO;
%end;

%end;

%if %length(&TopPercent) NE 0 %then %do;

DATA TOPPERCENTToChart_WithBarLabels;
length TargetSubTotal SubTotal 8;
retain TargetSubTotal SubTotal 0;
SET ALLtoChart_WithBarLabels;
if _N_ EQ 1
then TargetSubTotal = %sysevalf((&BarMeasureGrandTotal * &TopPercent) / 100);
SubTotal = SubTotal + &BarMeasureVar;
output;
if SubTotal GE TargetSubTotal;
call symput('TopPercentSubTotal',TRIM(LEFT(PUT(SubTotal,&BarMeasureFormat))));
call symput('TopPercentSubTotalPercent',TRIM(LEFT(PUT(( SubTotal /
&BarMeasureGrandTotal ) * 100 , 6.2 ))));
call symput('TopPercentSubTotalCount',TRIM(LEFT(PUT(_N_,3))));
STOP;
RUN;

%if &TopPercentSubTotalCount LE &MaximumBarCount
%then %let TopPercentChartPossible = YES;
%else %do;
%let TopPercent_Title1 = No subset of the data can be graphed if it requires more
than &MaximumBarCount bars;
%let TopPercent_Title2 = For TopPercent = &TopPercent, there are
&TopPercentSubTotalCount &Title1_Suffix that qualify;
%let TopPercentChartPossible = NO;
%end;

%end;

%if %length(&Min) NE 0 %then %do;

DATA GEMINToChart_WithBarLabels;
length SubTotal MinBarMeasureValue GeminCount 8;
retain SubTotal MinBarMeasureValue GeminCount 0;
SET ALLtoChart_WithBarLabels end=LastOne;
MinBarMeasureValue = &Min;
if &BarMeasureVar GE MinBarMeasureValue
then do;
output;
SubTotal = SubTotal + &BarMeasureVar;
GeminCount = _N_;
end;
if &BarMeasureVar LT MinBarMeasureValue
or
LastOne;
call symput('GeminSubTotal',TRIM(LEFT(PUT(SubTotal,&BarMeasureFormat))));

```

```

call symput('GeminSubTotalPercent',TRIM(LEFT(PUT(( SubTotal / &BarMeasureGrandTotal )
* 100 , 6.2))));
call symput('GeminSubTotalCount',TRIM(LEFT(PUT(GeminCount,3.))));
call symput('FormattedMin',TRIM(LEFT(PUT(MinBarMeasureValue,&BarMeasureFormat))));
STOP;
RUN;

%if &GeminSubTotalCount GT &MaximumBarCount %then %do;
%let Gemin_Title1 = No subset of the data can be graphed if it requires more than
&MaximumBarCount bars;
%let Gemin_Title2 = For Min = &Min, there are &GeminSubTotalCount &Title1_Suffix
that qualify;
%let GeminChartPossible = NO;
%end;
%else
%if &GeminSubTotalCount EQ 0 %then %do;
%let Gemin_Title1 = No subset of the data can be graphed;
%let Gemin_Title2 = For Min = &Min, there are no &Title1_Suffix that qualify;
%let GeminChartPossible = NO;
%end;
%else %let GeminChartPossible = YES;

%end;

%macro SubsetOneOfThreeWays_OR_All(Which=);

%let Which = %upcase(&Which);

%if &Which EQ TOPN AND &TopNChartPossible EQ YES
%then %do;
%let CountOfSelectedBars = &TopNSubTotalCount;
%end;
%else
%if &Which EQ TOPPERCENT AND &TopPercentChartPossible EQ YES
%then %do;
%let CountOfSelectedBars = &TopPercentSubTotalCount;
%end;
%else
%if &Which EQ GEMIN AND &GeminChartPossible EQ YES
%then %do;
%let CountOfSelectedBars = &GeminSubTotalCount;
%end;
%else %let CountOfSelectedBars = &CountOfAllBars;

%if %length(&TopN) NE 0 AND &TopNChartPossible EQ YES
%then %let SubsetMaxBarCount = &TopN;
%else %let SubsetMaxBarCount = 0;

%if %length(&TopPercent) NE 0 AND &TopPercentChartPossible EQ YES
%then %do;
%if &SubsetMaxBarCount LT &TopPercentSubTotalCount
%then %let SubsetMaxBarCount = &TopPercentSubTotalCount;
%end;

%if %length(&Min) NE 0 AND &GeminChartPossible EQ YES
%then %do;
%if &SubsetMaxBarCount LT &GeminSubTotalCount
%then %let SubsetMaxBarCount = &GeminSubTotalCount;
%end;

%if &SubsetMaxBarCount EQ 0 /* none of the three subset charts are possible */
%then %let SubsetMaxBarCount = %sysfunc(MIN( &CountOfAllBars , &MaximumBarCount ));
%put SubsetMaxBarCount = &SubsetMaxBarCount;

```

```

%if %eval(&MaximumBarCount EQ 50) AND %upcase(&For1024X768monitorsInstead) EQ NO
%then %do;

    %if %eval(&SubsetMaxBarCount LE 25)
    %then %do;
        %let TextFontPointSize = 14;
        %let YpixelsPerBar = 26;
    %end;
    %else
    %if %eval(&SubsetMaxBarCount LE 30)
    %then %do;
        %let TextFontPointSize = 12;
        %let YpixelsPerBar = 21.67;
    %end;
    %else
    %if %eval(&SubsetMaxBarCount LE 40)
    %then %do;
        %let TextFontPointSize = 8;
        %let YpixelsPerBar = 16.25;
    %end;
    %else
    %if %eval(&SubsetMaxBarCount LE 50)
    %then %do;
        %let TextFontPointSize = 6;
        %let YpixelsPerBar = 13;
    %end;
%end;

%end;

%else
%if %upcase(&For1024X768monitorsInstead) EQ YES
%then %do;

    %if %eval(&SubsetMaxBarCount LE &MaximumBarCount)
    %then %do;
        %let TextFontPointSize = &TextFontPtSizeFor1024X768monitor;
        %let YpixelsPerBar = &YpixelsPerBarFor1014X768monitor;
    %end;

%end;

%let BarChartPossible = YES;
%if &Which EQ ALL
%then %do;
    %let Title1 = &Title1_Prefix.%str( All )&CountOfSelectedBars &Title1_Suffix;
    %let Title2 = Selected All &Title1_Suffix;
%end;
%else
%if &Which EQ TOPN
%then %do;
    %if &TopNChartPossible EQ YES
    %then %do;
        %let Title1 = &Title1_Prefix.%str( Top )&CountOfSelectedBars &Title1_Suffix;
        %let Title2 = Selected Top &CountOfSelectedBars &Title1_Suffix
            - They Account for SubTotal &BarMeasureDescription &TopNSubTotal
            which is &TopNSubTotalPercent% of Grand Total;
        /* Because Title2 is delivered by HTML source code
        due to the specification of ODS HTML . . . NOGTITLE
        any extra blanks between the three parts of Title2
        on three separate lines in the %LET statement above
        will be compressed down to one blank. */
    %end;

```

```

%else %do;
    %let Title1 = &TopN_Title1;
    %let Title2 = &TopN_Title2;
    %let BarChartPossible = NO;
%end;
%end;
%else
%if &Which EQ TOPPERCENT
%then %do;
    %if &TopPercentChartPossible EQ YES
    %then %do;
        %let Title1 = &Title1_Prefix.%str( Top )&CountOfSelectedBars &Title1_Suffix;
        %let Title2 =
            Seeking At Least &TopPercent%, Top &CountOfSelectedBars &Title1_Suffix
            Account for SubTotal &BarMeasureDescription &TopPercentSubTotal
            which is &TopPercentSubTotalPercent% of Grand Total;
        /* Because Title2 is delivered by HTML source code
           due to the specification of ODS HTML . . . NOGTITLE
           any extra blanks between the three parts of Title2
           on three separate lines in the %LET statement above
           will be compressed down to one blank. */
    %end;
    %else %do;
        %let Title1 = &TopPercent_Title1;
        %let Title2 = &TopPercent_Title2;
        %let BarChartPossible = NO;
    %end;
%end;
%else
%if &Which EQ GEMIN
%then %do;
    %if &GEMinChartPossible EQ YES
    %then %do;
        %let Title1 = &Title1_Prefix.%str( Top )&CountOfSelectedBars &Title1_Suffix;
        %let Title2 = Selected Top &Title1_Suffix with &BarMeasureVar GE &FormattedMin
            - They Account for SubTotal &BarMeasureDescription &GEMinSubTotal
            which is &GEMinSubTotalPercent% of Grand Total;
        /* Because Title2 is delivered by HTML source code
           due to the specification of ODS HTML . . . NOGTITLE
           any extra blanks between the three parts of Title2
           on three separate lines in the %LET statement above
           will be compressed down to one blank. */
    %end;
    %else %do;
        %let Title1 = &GEMin_Title1;
        %let Title2 = &GEMin_Title2;
        %let BarChartPossible = NO;
    %end;
%end;

%if &BarChartPossible EQ YES
%then %let Y_pixels = %eval( %sysfunc(CEIL( %sysevalf(&CountOfSelectedBars *
&YpixelsPerBar) )) + &YpixelsOverhead );

%let Title3 = All &CountOfAllBars &Title1_Suffix Have Grand Total
&BarMeasureDescription &GrandTotal;

%if &UseTitle5ForDiagnostics EQ YES AND &BarChartPossible = YES %then %do;
    %let Title5ForDiagnostics =
        &CountOfSelectedBars bars, &YpixelsPerBar Ypixels per bar,
        &YpixelsOverhead Ypixels overhead, &Y_pixels Total Ypixels, &Xpixels Xpixels,
        Titles &TitleFontPointSize.pt &TitleFont, Bar Labels &TextFontPointSize.pt &TextFont;
    %let Title5ForDiagnostics = %sysfunc(compbl(%quote(&Title5ForDiagnostics)));

```

```

%end;
%else %let Title5ForDiagnostics = %str();
      /* so that any references to it resolve (to a null) */

%let FileNamePrefix = %sysfunc(COMPRESS(&Title1_Prefix,' '))&Title1_Suffix;

%let FileName = &FileNamePrefix._&Which.&FileNameDTsuffix;

FOOTNOTE;
ODS NORESULTS;
ODS LISTING CLOSE;

ODS HTML PATH="&RptPath" (URL=NONE) STYLE=Styles.MinimalWithNoFrame
  NOGTITLE NOGFOOTNOTE
  BODY="&FileName..html" (TITLE="&Title1");

TITLE1
%if &BarChartPossible = NO %then %do;
COLOR=RED
%end;
JUSTIFY=&TitlesJustify FONT=&TitleFont HEIGHT=&TitleFontPointSize PT "&Title1";
TITLE2
%if &BarChartPossible = NO %then %do;
COLOR=RED
%end;
JUSTIFY=&TitlesJustify FONT=&TitleFont HEIGHT=&TitleFontPointSize PT "&Title2";
TITLE3 JUSTIFY=&TitlesJustify FONT=&TitleFont HEIGHT=&TitleFontPointSize PT "&Title3";
TITLE4 JUSTIFY=&TitlesJustify FONT=&TitleFont HEIGHT=&TitleFontPointSize PT

%if %length(&TopN) NE 0
%then %do;
  LINK="&FileNamePrefix._TOPN&FileNameDTsuffix..html" "Go To Top &TopN"
%end;
%if %length(&TopPercent) NE 0
%then %do;
  LINK="&FileNamePrefix._TOPPERCENT&FileNameDTsuffix..html" "Go To Top &TopPercent% of
GrandTotal"
%end;
%if %length(&Min) NE 0
%then %do;
  LINK="&FileNamePrefix._GEMIN&FileNameDTsuffix..html" "Go To &BarMeasureVar GE &Min"
%end;
%if %upcase(&ShowAll) EQ YES
%then %do;
  LINK="&FileNamePrefix._ALL&FileNameDTsuffix..html" "Go To All"
%end;
%if &LinkableSpreadsheets EQ YES
%then %do;
  LINK="&FileName..XLS" "Spreadsheet of Input to this Chart"
%end;

;

%if %upcase(&UseTitle5ForDiagnostics) EQ YES
  OR
  %length(&Title5) NE 0
  OR
  %upcase(&RunDayDateTimeInTitle5) EQ YES
  OR
  %upcase(&LinkableSpreadsheets) EQ YES
%then %do;
TITLE5 JUSTIFY=&TitlesJustify FONT=&TitleFont HEIGHT=&TitleFontPointSize PT

```

```

%if %upcase(&UseTitle5ForDiagnostics) EQ YES %then %do;
COLOR=red HEIGHT=%eval(&TitleFontPointSize - 2) PT
/* make this long text string smaller */
"&Title5ForDiagnostics "
COLOR=black
%end;
%else
%if %length(&Title5) NE 0 %then %do;
COLOR=&Title5_Color
"&Title5 "
COLOR=black
%end;

%if %upcase(&LinkableSpreadSheets) EQ YES
%then %do;
    %if %length(&Title5) NE 0
        OR
        %length(&Title5ForDiagnostics) NE 0
    %then %do;
" | "
    %end;
COLOR=blue
"Click any bar for its detail spreadsheet "
COLOR=black
%end;

%if %upcase(&RunDayDateTimeInTitle5) EQ YES
%then %do;
    %if %length(&Title5) NE 0
        OR
        %length(&Title5ForDiagnostics) NE 0
        OR
        %upcase(&LinkableSpreadSheets) EQ YES
    %then %do;
" | "
    %end;
"&RunDayDateTime"
%end;

;
%end;

%if &BarChartPossible = YES
%then %do;

ODS GRAPHICS ON / RESET=ALL BORDER=&BorderOnOrOff SCALE=OFF
%if &LinkableSpreadSheets EQ YES
    OR
    &For1024X768monitorsInstead EQ NO
%then %do;
    IMAGEMAP=ON
%end;
    HEIGHT=&Y_pixels.px WIDTH=&Xpixels.px
    IMAGENAME="&FileName";

%if &LinkableSpreadSheets EQ YES %then %do;

DATA &Which.toChart_WithBarLabels;
LENGTH SpreadsheetFile $ 128;
SET &Which.toChart_WithBarLabels;
SpreadsheetFile = TRIM(LEFT(&BarLabelVar)) || "&FileNameDTsuffix..xls";
RUN;

```

```

%end;

PROC SGPLOT DATA=&Which.toChart_WithBarLabels DESCRIPTION='';
HBAR BarLabel / RESPONSE=&BarMeasureVar
%if &LinkableSpreadSheets EQ YES %then %do;
    URL=SpreadSheetFile
%end;
    CATEGORYORDER=RESPDESC BARWIDTH=1 FILL FILLATTRS=(COLOR=&BarColor) OUTLINE;
YAXIS DISPLAY=(NOLABEL NOLINE NOTICKS) VALUEATTRS=(SIZE=&TextFontPointSize PT);
XAXIS DISPLAY=NONE;
RUN;

%if &LinkableSpreadSheets EQ YES %then %do;

TITLE;
FOOTNOTE;
ODS NORESULTS;
ODS LISTING CLOSE;

/* Create the companion spreadsheet. */
/* GOPTIONS RESET=ALL;
Prior GOPTIONS can affect non-SAS/GRAPH PROC output. */
ODS HTML PATH="&RptPath" (URL=NONE) STYLE=Minimal
BODY="&FileName..XLS";
/* Use COLSPAN= below so that TITLE lines do not get put only into Column A and
expand its width to length of longest TITLE line.
Spread them across sixteen columns, which expand, if needed */

TITLE
JUSTIFY=LEFT "<td COLSPAN=16>&Title1</td>"
JUSTIFY=LEFT
"<td COLSPAN=16>
<a href='\" " &FileName..html" '\">
Go To Graph of This Data
</a>
</td>";

PROC PRINT DATA=&Which.toChart_WithBarLabels(DROP=SpreadSheetFile) LABEL NOOBS;
ID &BarLabelVar;
SUM &BarMeasureVar;
FORMAT &BarMeasureVar &BarMeasureFormat;
RUN;

%end;

ODS HTML CLOSE;
ODS LISTING;

%end;
%else %do;

PROC PRINT DATA=ALLtoChart_WithBarLabels NOOBS;
VAR BarLabel &BarMeasureVar;
RUN;

%end;

ODS HTML CLOSE;
ODS LISTING;

%mend SubsetOneOfThreeWays_OR_All;

%macro RemoveCraphAreaFrameFromStyle

```

```

(ParentStyle=,StyleWithNoFrame=);

PROC TEMPLATE;
DEFINE STYLE &StyleWithNoFrame;
  PARENT=&ParentStyle;
  CLASS GRAPHWALLS / FRAMEBORDER=OFF;
  /* remove a useless box around the bars */
END; RUN;
%mend RemoveCraphAreaFrameFromStyle;

%RemoveCraphAreaFrameFromStyle
(ParentStyle=Styles.Minimal
,StyleWithNoFrame=Styles.MinimalWithNoFrame);

%if %length(&ShowAll) NE 0 %then %do;
  %if %upcase(&ShowAll) EQ YES
    %then %SubsetOneOfThreeWays_OR_All(Which=All);
  %end;

%if %length(&TopN) NE 0
%then %SubsetOneOfThreeWays_OR_All(Which=TopN);

%if %length(&TopPercent) NE 0
%then %SubsetOneOfThreeWays_OR_All(Which=TopPercent);

%if %length(&Min) NE 0
%then %SubsetOneOfThreeWays_OR_All(Which=Gemin);

%if %upcase(&LinkableSpreadSheets) EQ YES %then %do;

proc sort data=&Data out=RankedWithinBarLabel;
by &BarLabelVar descending &BarMeasureVar;
run;

proc sort data=RankedWithinBarLabel(keep=&BarLabelVar) out=DistinctBarLabelValues
nodupkey;
by &BarLabelVar;
run;

data _null_;
set DistinctBarLabelValues end=LastOne;
call symput('BarLabel'|TRIM(LEFT(_N_)),TRIM(LEFT(&BarLabelVar)));
if LastOne;
call symput('BarLabelCount',_N_);
run;

%macro MakeSpreadSheets;

%do i = 1 %to &BarLabelCount %by 1;

TITLE;
FOOTNOTE;
ODS NORESULTS;
ODS LISTING CLOSE;

ODS HTML PATH="&RptPath" (URL=NONE) STYLE=Styles.Minimal
FILE="&&BarLabel&i..&FileNameDTSuffix..xls";

PROC PRINT NOOBS DATA=RankedWithinBarLabel(WHERE=(&BarLabelVar EQ "&&BarLabel&i"));
ID &BarLabelVar;
SUM &BarMeasureVar;
RUN;

```

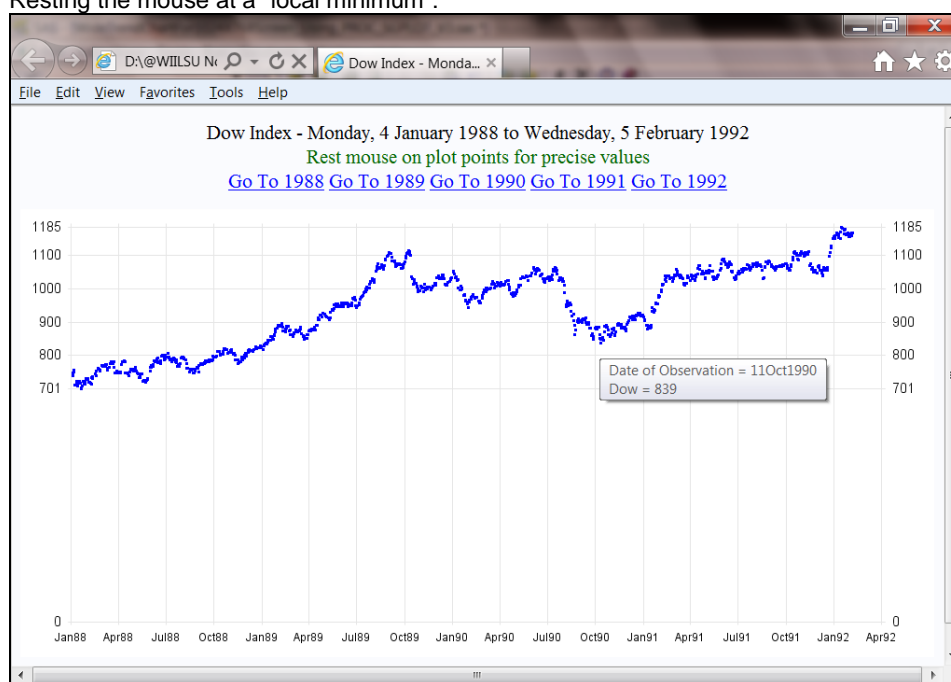


```
ODS HTML CLOSE;  
ODS LISTING;  
  
%end;  
  
%mend  MakeSpreadSheets;  
  
%MakeSpreadSheets;  
  
%end;  
  
%MacExit:  
  
%mend  UpTo4WaysRankingHbarChartsToHTML;
```

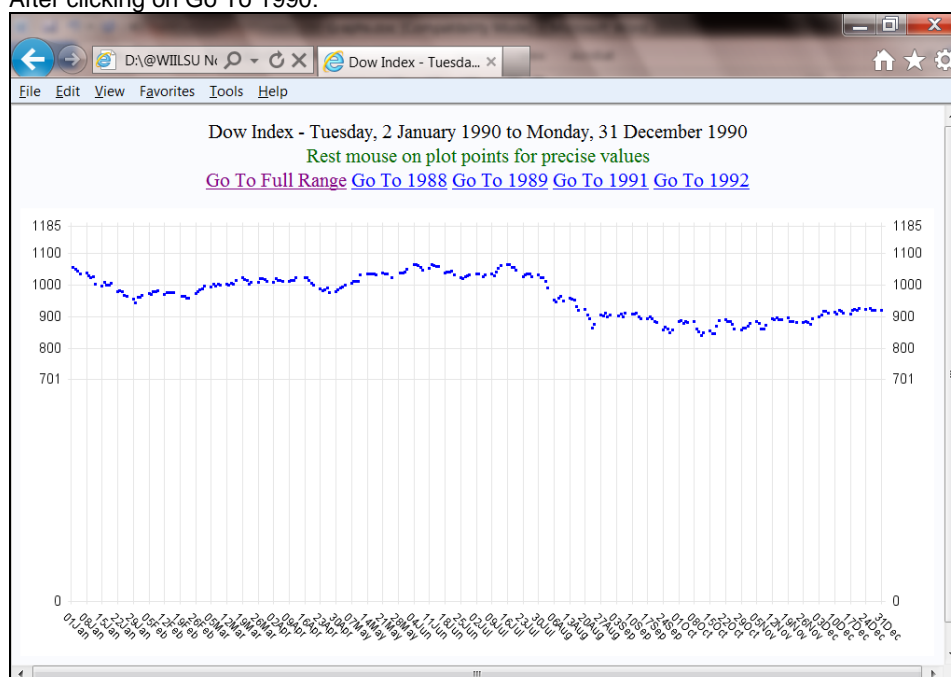
TOOL 6. ODS GRAPHICS PROC SGPLOT SOLUTION FOR VERY LONG TREND CHARTS

One would like to see an overview of the total time period. However, since showing that time period requires a comparatively sparse set of horizontal axis tick mark values, it is helpful to be able to zoom in on sub-periods of the total period. Below is a solution that I like. Web deployment for trend data, with the availability of ALT text, is always preferable to static delivery. **The title above is in red because, until an anomaly in ODS Graphics is addressed, this solution is not recommended for reasons explained below.** When that problem is fixed, the solution presented here would be recommended. Because it is not, it has not been fully packaged as a single invocable macro and optional linkages to spreadsheets of the plot input data are not provided. **For now, Tool 7 is the better choice.**

Resting the mouse at a “local minimum”:



After clicking on Go To 1990:



The graphs above start the y axis at zero to avoid magnifying minor fluctuations, and maintain the same y tick mark values and reference lines across all views for easy comparability. When moving the cursor around the web page, but not resting it on a plot point, there is displayed a default description of the graph provided by ODS Graphics. Unlike SAS/GRAPH, there is no way to customize it, and no way to suppress it.

The title lines for the graphs above must be delivered outside the image file (using ODS HTML . . . NOGTITLE) so that the hyperlinks will work. (This is not the situation for SAS/GRAPH PROC GPLOT.) With the ODS Graphics SG procedures there is no direct control of the GRID (reference lines). Unlike the case with the SAS/GRAPH GPLOT procedure, where reference lines can be specified at selected tick mark values OR turned on for all major tick marks (this latter alternative being analogous to GRID in ODS Graphics) and where line type and line color can be controlled, the ODS Graphics grid characteristics are controlled by whatever ODS Style has been selected. Here the HTMLBLUE style was used. If you look closely, you can see that the graph image file is surrounded by a light blue background, and above the image file it is overlaid with title lines (due to the NOGTITLE option mentioned above).

NOTE: With the ODS Graphics PROC SPLOT SERIES statement the IMAGEMAP (HTML source code) that defines the hot spots for the cursor to display the ALT text is created in an unnecessarily inefficient manner. The boundary of each hot spot is defined three times for the marker being drawn for the left-hand vertical axis, and each of the three boundary definitions is different, although they are not extremely different. Furthermore, in order to get the tick mark values displayed on the right-hand vertical axis the markers must be drawn a second time, and the imagemap contains another three boundary definitions for each hot spot. The second set of three different boundary definitions DOES match the first set. The size of the HTML file for the full date range graph is 1058 KB, and for the one-year date range graphs varies, but is approximately 283 KB. The IMAGEMAP drawn for SAS/GRAPH PROC GPLOT is efficient and much smaller. Each hot spot is defined only once, even when both vertical axes have tick mark values. The HTML files created with SAS/GRAPH PROC GPLOT are 95 KB and 26 KB—i.e., less than one-tenth the size of those created with ODS Graphics PROC SGPLOT. This concern about efficient definition of IMAGEMAP is not simply an obsession with disk space to store the web page or with file download time when opening the web page. In a different context, when creating a set of several point-dense multi-line plots, my users and I found that opening several at the same time would eventually render them unusable, as well as rendering other applications in other open windows unusable. The solution was to select a different plot point marker for which SAS/GRAPH would create more efficient IMAGEMAP code. See, e.g., the comments for SYMBOL statements in the Tool 8 macro. For more information, send an email to me. Problems caused by inefficient IMAGEMAP construction might be the reason why the ODS GRAPHICS statement includes a TIPMAX option to specify the maximum number of hot spots allowed before the ALT text is all disabled.

Here is the code used to create the above graphs with ODS Graphics and PROC SGPLOT:

```
libname CITIHELP "D:\SASHELP Sample data sets from SAS ETS";

DATA WORK.DowByDay;
LABEL snydjcm='Dow';
LENGTH Year $4;
SET CITIHELP.citiday(KEEP=date snydjcm WHERE=(snydjcm NE .));
snydjcm = ROUND(snydjcm,1);
Year = YEAR(date);
RUN;

/* START of SetUp Code not unique to use of ODS Graphics */

PROC MEANS DATA=WORK.DowByDay MIN MAX NOPRINT;
VAR snydjcm date;
OUTPUT OUT=WORK.MinMax MIN=MinY StartDate MAX=MaxY EndDate;
RUN;

DATA _NULL_;
SET MinMax;
CALL SYMPUT('MinY',TRIM(LEFT(MinY)));
CALL SYMPUT('MaxY',TRIM(LEFT(MaxY)));
CALL SYMPUT('DateRange',
    TRIM(LEFT(PUT(StartDate,WEEKDATX32.))) ||
    ' to ' ||
    TRIM(LEFT(PUT(EndDate, WEEKDATX32.))));
RUN;
```

```

PROC SUMMARY DATA=WORK.DowByDay NWAY;
CLASS Year;
VAR date;
OUTPUT OUT=WORK.EachYearWithStartDateEndDate MIN=StartDate MAX=EndDate;
RUN;

DATA _NULL_;
SET WORK.EachYearWithStartDateEndDate END=LastYear;
CALL SYMPUT('DateRange' || TRIM(LEFT(_N_)),
            TRIM(LEFT(PUT(StartDate, WEEKDATX32.))
                || ' to ' ||
                TRIM(LEFT(PUT(EndDate, WEEKDATX32.))));
CALL SYMPUT('Year' || TRIM(LEFT(_N_)), TRIM(LEFT(Year)));
IF LastYear;
CALL SYMPUT('NumberOfYears', TRIM(LEFT(_N_)));
RUN;

DATA _NULL_;
SET WORK.EachYearWithStartDateEndDate END=LastYear;
* Want to start on a Monday not later than the first day of data *;
StartDate = MDY(1,1,Year);
DayOfWeekStart = WEEKDAY(StartDate);
IF DayOfWeekStart GT 2
THEN StartDate = StartDate - (DayOfWeekStart - 2);
ELSE
IF DayOfWeekStart EQ 1
THEN StartDate = StartDate - 6;
CALL SYMPUT('StartDateTick' || TRIM(LEFT(_N_)), StartDate);
* Want to end on a Monday not earlier than the last day of data *;
EndDate = MDY(12,31,Year);
DayOfWeekEnd = WEEKDAY(EndDate);
IF DayOfWeekEnd GT 2
THEN EndDate = EndDate + (9 - DayOfWeekEnd);
ELSE
IF DayOfWeekEnd EQ 1
THEN EndDate = EndDate + 1;
CALL SYMPUT('EndDateTick' || TRIM(LEFT(_N_)), EndDate);
RUN;

/* End of SetUp Code not unique to use of ODS Graphics */

%let Path = D:\!PharmasUG2013\Graphs\ResultsToHTML\WidePlots\Using_PROC_SGPLOT;

%MACRO CreateYearLinks_SGPLOT;

%DO j = 1 %TO &NumberOfYears %BY 1;
    LINK="Year_&&Year&j...html" "Go To &&Year&j"
%END;

%MEND CreateYearLinks_SGPLOT;

PROC TEMPLATE;
DEFINE STYLE styles.htmlblueWithNoFrame; /* remove useless box around the plot area */
    PARENT=Styles.htmlblue;
    CLASS graphwalls / frameborder=off;
END; RUN;

ODS NORESULTS; /* do not open in SAS session */
ODS LISTING CLOSE;

/* START SetUp that applies to all graphs */

```

```

GOPTIONS RESET=ALL;
ODS GRAPHICS ON / RESET=ALL BORDER=OFF IMAGEMAP TIPMAX=2500 ANTIALIASMAX=2500
    WIDTH=980px HEIGHT=480px; /* override default size 480px X 360px */

/* END SetUp that applies to all graphs */

/* HEIGHT for TITLES rendered with ODS NOGTITLE cannot use PCT as the Unit */
TITLE1 FONT='Times New Roman' HEIGHT=16pt COLOR=black
    "Dow Index - &DateRange" ;
TITLE2 FONT='Times New Roman' HEIGHT=16pt COLOR=CX006600
    "Rest mouse on plot points for precise values";
TITLE3 FONT='Times New Roman' HEIGHT=16pt
    %CreateYearLinks_SGPLOT
    ;

ODS HTML PATH="&Path" (URL=NONE) STYLE=Styles.htmlblueWithNoFrame NOGTITLE GFOOTNOTE
    BODY="FullRange_QuarterlyHticks.html" (TITLE="Dow Index - &DateRange");

ODS GRAPHICS ON / MAXLEGENDAREA=0 /* suppress legend caused by extra Y2AXIS */
    IMAGENAME="atZERO";

PROC SGPLOT DATA=WORK.DowByDay;
SERIES Y=snydjcm X=date /
    MARKERS MARKERATTRS=(SIZE=3 SYMBOL=circlefilled COLOR=blue)
    LINEATTRS=(COLOR=white);
SERIES Y=snydjcm X=date / Y2AXIS /* create a right-hand y axis */
    MARKERS MARKERATTRS=(SIZE=3 SYMBOL=circlefilled COLOR=blue)
    LINEATTRS=(COLOR=white);
YAXIS DISPLAY=(NOLABEL NOLINE NOTICKS) GRID
    VALUEATTRS=(SIZE=2PCT FAMILY='Arial')
    MIN=0 MAX=&MaxY VALUESHINT VALUES=(0 &MinY 800 900 1000 1100 &MaxY);
Y2AXIS /* define the right-hand y axis */
    DISPLAY=(NOLABEL NOLINE NOTICKS) GRID
    VALUEATTRS=(SIZE=2PCT FAMILY='Arial')
    MIN=0 MAX=&MaxY VALUESHINT VALUES=(0 &MinY 800 900 1000 1100 &MaxY);
XAXIS INTERVAL=quarter FITPOLICY=ROTATE
    DISPLAY=(NOLABEL NOLINE NOTICKS) GRID
    VALUEATTRS=(SIZE=1.5PCT FAMILY='Arial') TICKVALUEFORMAT=MONYY5.;
RUN;

ODS HTML CLOSE;

%MACRO LinkedYearlyTrendCharts_SGPLOT;

%DO i = 1 %TO &NumberOfYears %BY 1;

    /* HEIGHT for TITLES rendered with ODS NOGTITLE cannot use PCT as the Unit */
    TITLE1 FONT='Times New Roman' HEIGHT=16pt COLOR=black "Dow Index - &&DateRange&i" ;
    TITLE2 FONT='Times New Roman' HEIGHT=16pt COLOR=CX006600 "Rest mouse on plot points
    for precise values";
    TITLE3 FONT='Times New Roman' HEIGHT=16pt
        LINK="FullRange_QuarterlyHticks.html" "Go To Full Range"
    %DO j = 1 %TO &NumberOfYears %BY 1;
        %IF &&Year&j NE &&Year&i
        %THEN %DO;
            LINK="Year_&&Year&j...html" "Go To &&Year&j"
        %END;
    %END;
    ;

ODS HTML PATH="&Path" (URL=NONE) STYLE=Styles.htmlblueWithNoFrame NOGTITLE GFOOTNOTE
    BODY="Year_&&Year&i...html" (TITLE="Dow Index - &&DateRange&i");

```

```

ODS GRAPHICS ON / IMAGENAME="atZERO&i" ;

PROC SGPLOT DATA=WORK.DowByDay(WHERE=(Year EQ "&&Year&i" ));
SERIES Y=snydjcm X=date /
    LINEATTRS=(COLOR=white)
    MARKERS MARKERATTRS=(SIZE=3 SYMBOL=circlefilled COLOR=blue);
/* The default unit for SIZE is px (pixels). At a size of only three pixels
the symbol which should be a circle can be drawn only as a square.
Because there are so many plot points, if they are to be distinguishable,
it is necessary to render them at a small size. */
SERIES Y=snydjcm X=date / Y2AXIS /* create a right-hand y axis */
    LINEATTRS=(COLOR=white) /* there is no way to suppress the unneeded line */
    MARKERS MARKERATTRS=(SIZE=3 SYMBOL=circlefilled COLOR=blue);
/* Using the NOMARKERS option to suppress the second drawing of the markers,
and thus use Y2AXIS only to get the tick mark labels at the right-hand side
would have the undesirable effect of having the white plot line for the
right-hand y axis drawn over the blue markers that are first written over
the white plot line for the left-hand y axis. */
YAXIS DISPLAY=(NOLABEL NOLINE NOTICKS) GRID
    VALUEATTRS=(SIZE=2PCT FAMILY='Arial')
    MIN=0 MAX=&MaxY VALUESHINT VALUES=(0 &MinY 800 900 1000 1100 &MaxY);
Y2AXIS /* define the right-hand y axis */
    DISPLAY=(NOLABEL NOLINE NOTICKS) GRID
    VALUEATTRS=(SIZE=2PCT FAMILY='Arial')
    MIN=0 MAX=&MaxY VALUESHINT VALUES=(0 &MinY 800 900 1000 1100 &MaxY);
XAXIS DISPLAY=(NOLABEL NOLINE NOTICKS) GRID
    VALUEATTRS=(SIZE=1.5PCT FAMILY='Arial') TICKVALUEFORMAT=DATE5.
    VALUES=(&&StartDateTick&i to &&EndDateTick&i by 7)
    FITPOLICY=ROTATE;
RUN;
ODS HTML CLOSE;

%END;

%MEND LinkedYearlyTrendCharts_SGPLOT;

OPTIONS MPRINT;

%LinkedYearlyTrendCharts_SGPLOT;

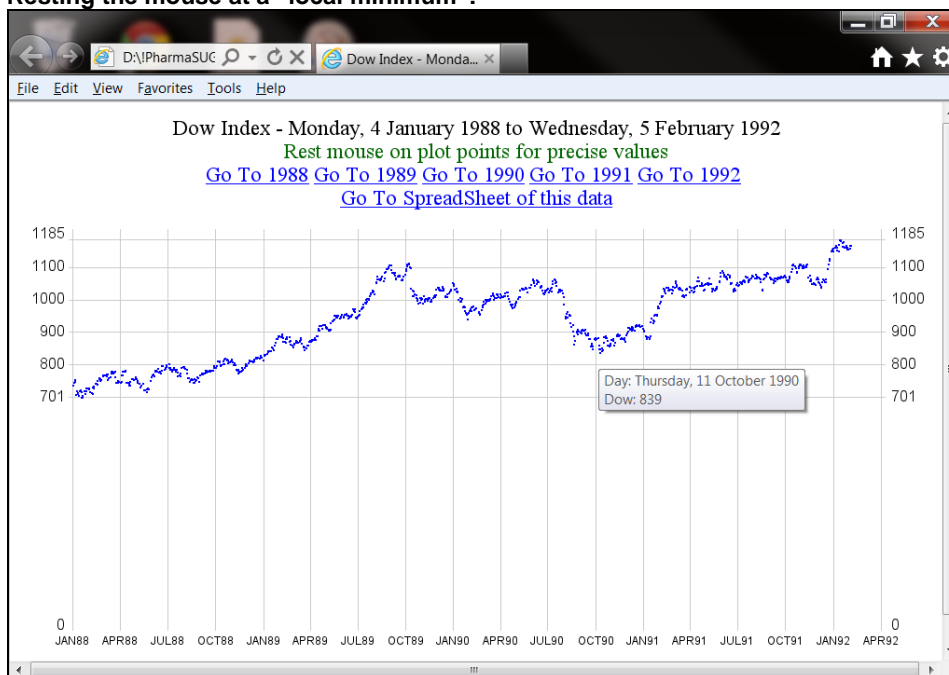
ODS LISTING;

```

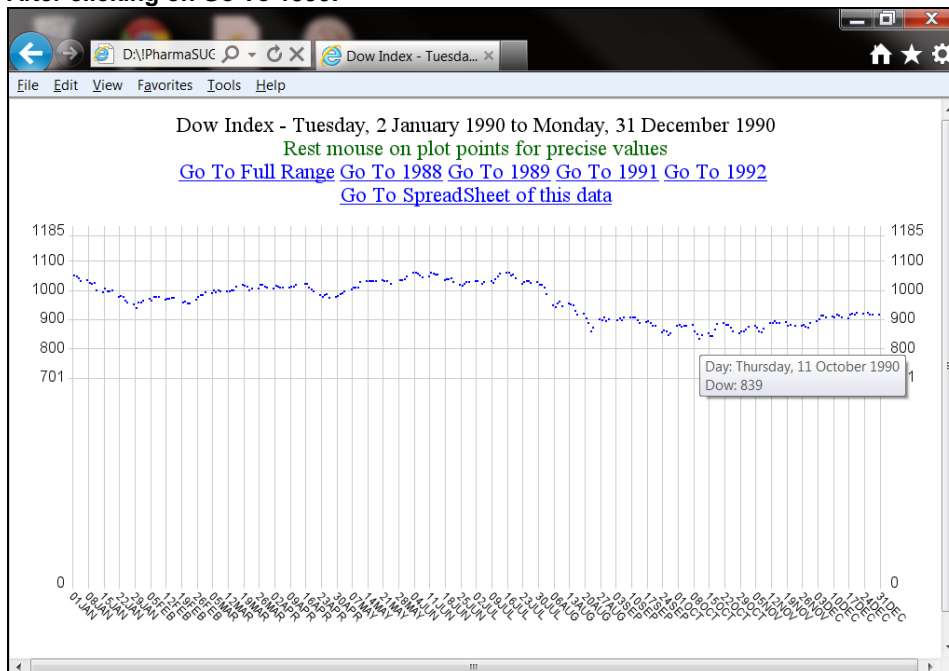
TOOL 7. SAS/GRAPH PROC GPLOT SOLUTION FOR VERY LONG TREND CHARTS

Here is the same pair of plots created with SAS/GRAPH PROC GPLOT. **Note the custom ALT text.**

Resting the mouse at a “local minimum”:



After clicking on Go To 1990:



Because the links above are in a title line that overlays the image file (due to ODS HTML . . . GTITLE), they do not change color after having been visited.

With SAS/GRAPH PROC GPLOT the IMAGEMAP (HTML source code) that defines the hot spots for the cursor to display the ALT text is created very efficiently. The sizes of the HTML file for the full date range graph is 95 KB and

for the one-year date range graphs is 26 KB. The sizes of the corresponding HTML files created with ODS Graphics PROC SGPLOT are 1058 KB and 283 KB—i.e., over ten times the size of those created with SAS/GRAPH GPLOT. Below is the macro and its invocation code to create the plots above.

NOTE: When running the code below, the steps that create the one-year-at-a-time plots generate a misleading WARNING message that says (wrongly): The bottom horizontal axis labeled Date of Observation could not be fit as specified. The axis values will overwrite.

```
%MACRO Wide_Plots_Using_PROC_GPLOT
(Path=
,Data=
,LinkableSpreadSheets=NO
);

%MACRO CreateYearLinks_GPLOT;
%DO j = 1 %TO &NumberOfYears %BY 1;
  LINK="Year_&&Year&j...html" UNDERLINE=1 "Go To &&Year&j" UNDERLINE=0 " "
%END;
%MEND CreateYearLinks_GPLOT;

/* START of SetUp Code not unique to use of SAS/GRAPH PROC GPLOT */

PROC MEANS DATA=&Data MIN MAX NOPRINT;
VAR snydjcm date;
OUTPUT OUT=WORK.MinMax MIN=MinY StartDate MAX=MaxY EndDate;
RUN;

DATA _NULL_;
SET MinMax;
CALL SYMPUT('MinY',TRIM(LEFT(MinY)));
CALL SYMPUT('MaxY',TRIM(LEFT(MaxY)));
CALL SYMPUT('DateRange',
  TRIM(LEFT(PUT(StartDate,WEEKDATX32.))) ||
  ' to ' ||
  TRIM(LEFT(PUT(EndDate, WEEKDATX32.))));
RUN;

PROC SUMMARY DATA=&Data NWAY;
CLASS Year;
VAR date;
OUTPUT OUT=WORK.EachYearWithStartDateEndDate MIN=StartDate MAX=EndDate;
RUN;

DATA _NULL_;
SET WORK.EachYearWithStartDateEndDate END=LastYear;
CALL SYMPUT('DateRange' || TRIM(LEFT(_N_)),
  TRIM(LEFT(PUT(StartDate, WEEKDATX32.)))
  || ' to ' ||
  TRIM(LEFT(PUT(EndDate, WEEKDATX32.))));
CALL SYMPUT('Year' || TRIM(LEFT(_N_)),TRIM(LEFT(Year)));
IF LastYear;
CALL SYMPUT('NumberOfYears',TRIM(LEFT(_N_)));
RUN;

DATA _NULL_;
SET WORK.EachYearWithStartDateEndDate END=LastYear;
* Want to start on a Monday not later than the first day of data *;
StartDate = MDY(1,1,Year);
DayOfWeekStart = WEEKDAY(StartDate);
IF DayOfWeekStart GT 2
THEN StartDate = StartDate - (DayOfWeekStart - 2);
ELSE
IF DayOfWeekStart EQ 1
```



```

THEN StartDate = StartDate - 6;
CALL SYMPUT('StartDateTick' || TRIM(LEFT(_N_)), StartDate);
* Want to end on a Monday not earlier than the last day of data */
EndDate = MDY(12, 31, Year);
DayOfWeekEnd = WEEKDAY(EndDate);
IF DayOfWeekEnd GT 2
THEN EndDate = EndDate + (9 - DayOfWeekEnd);
ELSE
IF DayOfWeekEnd EQ 1
THEN EndDate = EndDate + 1;
CALL SYMPUT('EndDateTick' || TRIM(LEFT(_N_)), EndDate);
RUN;

/* End of SetUp Code not unique to use of SAS/GRAPH PROC GPLOT */

DATA WORK.ToPlot;
LENGTH HTMLvar $ 512;
SET WORK.DowByDay;
HTMLvar = "alt="
|| "Day: " || TRIM(LEFT(PUT(Date, WEEKDATX32.)))
|| '0D'X /* force a line break in the ALT text display */
|| "Dow: " || TRIM(LEFT(PUT(snydjcm, 4.))) || " ";
RUN;

PROC CATALOG CAT=WORK.gseg KILL; RUN; QUIT;

ODS NORESULTS; /* do not open in SAS session */
ODS LISTING CLOSE;

/* START SetUp that applies to all graphs */

ODS GRAPHICS OFF;
GOPTIONS RESET=ALL;
GOPTIONS FTEXT="Arial"; /* for axis values */
GOPTIONS XPIXELS=980 YPIXELS=576;

SYMBOL1 COLOR=blue HEIGHT=0.5 FONT=ZAPF VALUE='E2'X;
SYMBOL2 COLOR=blue HEIGHT=0.5 FONT=ZAPF VALUE='E2'X;
/* Use of FONT=ZAPF VALUE='E2'X results in a smaller HTML file
because many other SAS/GRAPH symbol choices cause the AREA map, used to provide
ALT text, to be generated in a very inefficient manner. When multiple browse
windows are open, not only those browser windows, but also other windows that
the viewing user might have open, can become inoperative. That problem of huge
area maps might have been observed only with very large numbers of plot points
on multi-line plots. Multiple lines multiply the numbers of plots points.
Instead of FONT=ZAPF VALUE='E2'X, an unwise symbol choice for a very dense
multi-line plot would be, e.g., VALUE=DOT. */

AXIS1 LABEL=NONE MAJOR=NONE MINOR=NONE STYLE=0 /* left-hand y axis */
ORDER = 0 TO 1200 BY 100
VALUE=(HEIGHT=3PCT
'0' ' ' ' ' ' ' ' ' ' ' "&MinY" '800' '900' '1000' '1100' "&MaxY");

AXIS2 LABEL=NONE MAJOR=NONE MINOR=NONE STYLE=0 /* right-hand y axis */
ORDER = 0 TO 1200 BY 100
VALUE=( HEIGHT=3PCT JUSTIFY=LEFT
'0' ' ' ' ' ' ' ' ' ' ' "&MinY" '800' '900' '1000' '1100' "&MaxY");

/* END SetUp that applies to all graphs */

AXIS3 LABEL=NONE MAJOR=NONE MINOR=NONE STYLE=0 VALUE=(HEIGHT=2PCT);

TITLE1 FONT='Times New Roman' HEIGHT=4PCT COLOR=black

```

```

        "Dow Index - &DateRange";
TITLE2 FONT='Times New Roman' HEIGHT=4PCT COLOR=CX006600
        "Rest mouse on plot points for precise values";
TITLE3 HEIGHT=4PCT FONT='Times New Roman'
        COLOR=blue %CreateYearLinks_GPLOT
        ;
%IF %upcase(&LinkableSpreadSheets) EQ YES %THEN %DO;
TITLE4 HEIGHT=4PCT FONT='Times New Roman' COLOR=blue
        LINK="FullRange.XLS" UNDERLINE=1 "Go To Spreadsheet of this data";
%END;

ODS HTML PATH="&Path" (URL=NONE) STYLE=Styles.Minimal GTITLE GFOOTNOTE
        BODY="FullRange_DefaultHticks.html" (TITLE="Dow Index - &DateRange");

PROC GPLOT DATA=WORK.ToPlot;
PLOT snydjcm * date / HTML=HTMLvar NAME="atZERO" DESCRIPTION=' '
        VAXIS=AXIS1 CVREF=CXCCCCC LVREF=1 VREF=(&MinY 800 900 1000 1100 &MaxY)
        HAXIS=AXIS3 AUTOHREF CHREF=CXCCCCC LHREF=1;
PLOT2 snydjcm*date / VAXIS=AXIS2;
FORMAT date MONYY5.;
RUN; QUIT;

ODS HTML CLOSE;

%IF %upcase(&LinkableSpreadSheets) EQ YES %THEN %DO;

        /* Create the companion spreadsheet. */
        /* GOPTIONS RESET=ALL;
        Prior GOPTIONS can affect non-SAS/GRAPH PROC output. */
ODS HTML PATH="&Path" (URL=NONE) STYLE=Minimal
        BODY="FullRange.XLS";
        /* Use COLSPAN= below so that TITLE lines do not get put only into Column A and
        expand its width to length of longest TITLE line.
        Spread them across sixteen columns, which expand, if needed */
TITLE
        JUSTIFY=LEFT "<td COLSPAN=16>Dow Index - &DateRange</td>"
        JUSTIFY=LEFT
        "<td COLSPAN=16>
        <a href='\" "FullRange_DefaultHticks.html" '\">
        Go To Graph of This Data
        </a>
        </td>";
PROC PRINT DATA=WORK.ToPlot(DROP=HTMLvar) LABEL NOOBS;
LABEL snydjcm='Dow';
FORMAT date WEEKDATX32.;
FORMAT snydjcm 4.;
RUN;
ODS HTML CLOSE;

%END;

%MACRO LinkedYearlyTrendCharts_GPLOT;

%DO i = 1 %TO &NumberOfYears %BY 1;

AXIS3 LABEL=NONE MAJOR=NONE MINOR=NONE STYLE=0 VALUE=(HEIGHT=2PCT ANGLE=-45)
        ORDER = &&StartDateTick&i to &&EndDateTick&i by 7;
TITLE1 FONT='Times New Roman' HEIGHT=4PCT COLOR=black
        "Dow Index - &&DateRange&i";
TITLE2 FONT='Times New Roman' HEIGHT=4PCT COLOR=CX006600
        "Rest mouse on plot points for precise values";
TITLE3 HEIGHT=4PCT FONT='Times New Roman' COLOR=blue
        LINK="FullRange_DefaultHticks.html" UNDERLINE=1 "Go To Full Range" UNDERLINE=0 " "

```

```

%DO j = 1 %TO &NumberOfYears %BY 1;
  %IF &&Year&j NE &&Year&i
    %THEN %DO;
      LINK="Year_&&Year&j...html" UNDERLINE=1 "Go To &&Year&j" UNDERLINE=0 " "
    %END;
  %END;
;
%IF %upcase(&LinkableSpreadSheets) EQ YES %THEN %DO;
TITLE4 HEIGHT=4PCT FONT='Times New Roman' COLOR=blue
  LINK="Year_&&Year&i...XLS" UNDERLINE=1 "Go To SpreadSheet of this data";
%END;

ODS HTML PATH="&Path" (URL=NONE) STYLE=Minimal GTITLE GFOOTNOTE
  BODY="Year_&&Year&i...html" (TITLE="Dow Index - &&DateRange&i");
PROC GPLOT DATA=WORK.ToPlot(WHERE=(Year EQ "&&Year&i"));
PLOT snydjcm * date / HTML=HTMLvar NAME="atZERO&i" DESCRIPTION=' '
  VAXIS=AXIS1 CVREF=CXCCCCC LVREF=1 VREF=(&MinY 800 900 1000 1100 &MaxY)
  HAXIS=AXIS3 AUTOHREF CHREF=CXCCCCC LHREF=1;
PLOT2 snydjcm * date / VAXIS=AXIS2;
FORMAT date DATE5.;
RUN; QUIT;
ODS HTML CLOSE;

  %IF %upcase(&LinkableSpreadSheets) EQ YES %THEN %DO;

/* Create the companion spreadsheet. */
/* GOPTIONS RESET=ALL;
  Prior GOPTIONS can affect non-SAS/GRAPH PROC output. */
ODS HTML PATH="&Path" (URL=NONE) STYLE=Minimal
  BODY="Year_&&Year&i...XLS";
/* Use COLSPAN= below so that TITLE lines do not get put only into Column A and
  expand its width to length of longest TITLE line.
  Spread them across sixteen columns, which expand, if needed */
TITLE
  JUSTIFY=LEFT "<td COLSPAN=16>Dow Index - &&DateRange&i</td>"
  JUSTIFY=LEFT
  "<td COLSPAN=16>
  <a href='\" "Year_&&Year&i...html" '\">
  Go To Graph of This Data
  </a>
  </td>";
PROC PRINT DATA=WORK.ToPlot(WHERE=(Year EQ "&&Year&i") DROP=HTMLvar) LABEL NOOBS;
LABEL snydjcm='Dow';
FORMAT date WEEKDATX32.;
FORMAT snydjcm 4.;
RUN;
ODS HTML CLOSE;

  %END;

%END;

%MEND LinkedYearlyTrendCharts_GPLOT;

OPTIONS MPRINT;

%LinkedYearlyTrendCharts_GPLOT;

ODS LISTING;

%MEND Wide_Plots_Using_PROC_GPLOT;

libname CITIHELP "D:\SASHELP Sample data sets from SAS ETS";

```

```

/* There are five cyclic data sets of financial and economic data,
   citiday, citiwk, citimon, citiqtr, and citiyr,
   that are shipped when your site licenses SAS/ETS.
   If you do not have SAS/ETS, SAS Technical Support can tell you
   how to download them. */

```

```

DATA WORK.DowByDay;
LABEL snydjcm='Dow';
LENGTH Year $4;
SET CITIHELP.citiday(KEEP=date snydjcm WHERE=(snydjcm NE .));
snydjcm = ROUND(snydjcm,1);
Year = YEAR(date);
RUN;

```

```

%Wide_Plots_Using_PROC_GPLOT
(Path=D:\!PharmaSUG2013\Graphs\ResultsToHTML\WidePlots\Using_PROC_GPLOT
,Data=work.DowByDay
,LinkableSpreadSheets=YES);

```

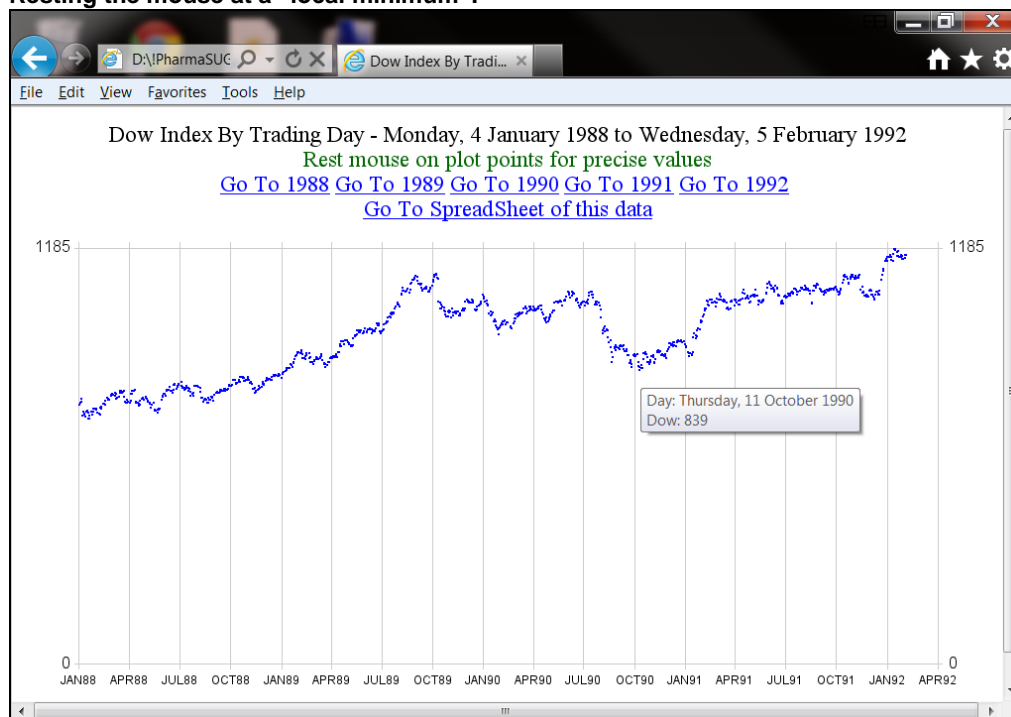
The above macro contains a lot of hard code specific to the characteristics of the input data set used. Next, let's look at a macro that is intended to be more general purpose. The price of being non-specific is the inability to customize the intermediate vertical tick marks and their reference lines. However, since ALT text is available, and delivers PRECISE y values, what is lost are only customary graphic features, not genuinely communication-effective features.

TOOL 8. SAS/GRAPH GPLOT MACRO FOR ANY MULTI-YEAR DAILY DATA MEASURE

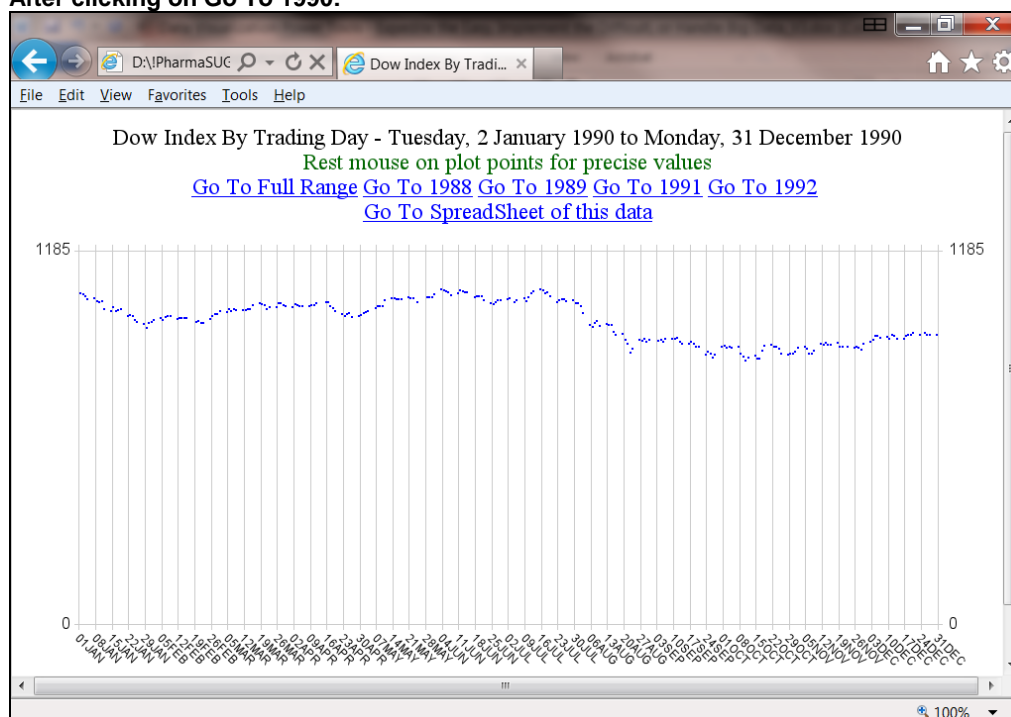
The macro can use the y-axis vertical space maximally, which can misleadingly amplify minor increases and decreases, or can force the y-axis to start at zero and minimize the visual impact of fluctuations.

Here are a pair of plots, using the ForceVzero=YES option.

Resting the mouse at a “local minimum”:

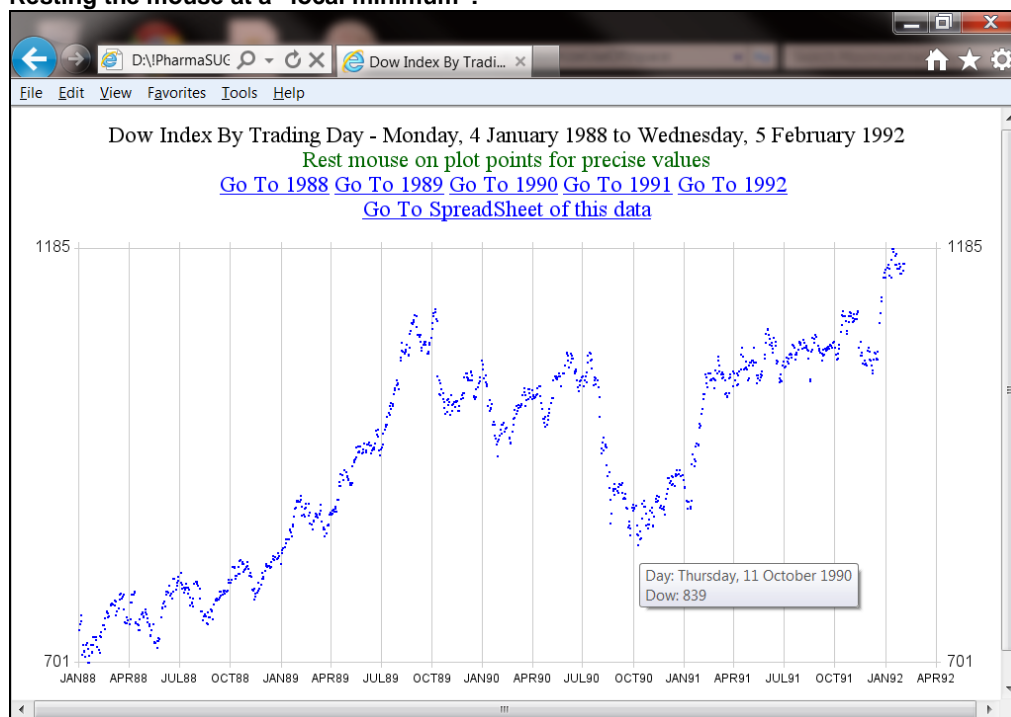


After clicking on Go To 1990:

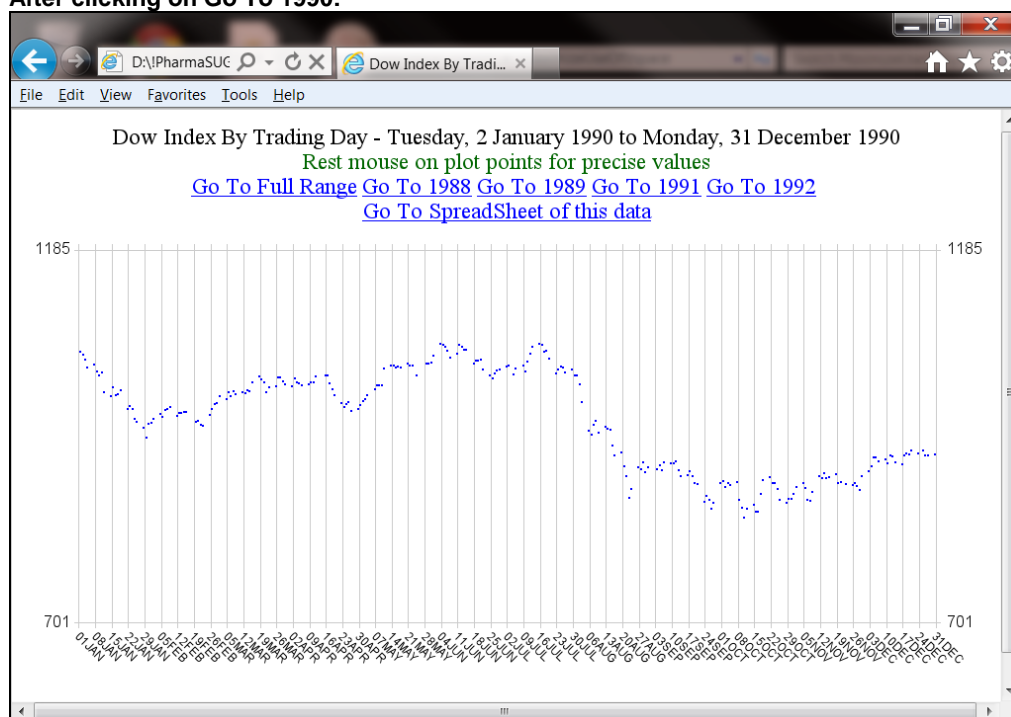


Here is the same pair of plots, but using the ForceVzero=NO option.

Resting the mouse at a “local minimum”:



After clicking on Go To 1990:



As noted earlier, with pop-up ALT text available, the absence of intermediate tick mark values and reference lines is NOT really a loss in communication effectiveness. Precise values are reliable, visual estimates of plot point values by comparison with tick marks and reference lines are not.

Below are a preparatory DATA step and the macro invocations used to create the plots, followed by the macro itself.

```
libname CITIHELP "D:\SASHELP Sample data sets from SAS ETS";

/* There are five cyclic data sets of financial and economic data,
   citiday, citiwk, citimon, citiqtr, and citiyr,
   that are shipped when your site licenses SAS/ETS.
   If you do not have SAS/ETS, SAS Technical Support can tell you
   how to download them. */

DATA WORK.DowByDay;
LABEL snydjcm='Dow';
LENGTH Year $4;
SET CITIHELP.citiday(KEEP=date snydjcm WHERE=(snydjcm NE .));
snydjcm = ROUND(snydjcm,1);
Year = YEAR(date);
RUN;

%Wide_Plots_PROC_GPLOTT_ForAnyData
(Path=D:\!PharmaSUG2013\Graphs\ResultsToHTML\WidePlots\Using_PROC_GPLOTT\ForceVzero
,Data=work.DowByDay
,Yvar=snydjcm
,YvarFormat=4.
,YvarLabel=Dow
,Xvar=date
,XvarFormatForFullRange=MONYY5.
,XvarFormatForSubRange=DATE5.
,XvarFormatForALTtext=WEEKDATX32.
,XvarLabel=Day
,DateRangeFormat=WEEKDATX32.
,LinkableSpreadSheets=YES
,SpreadSheetYvarFormat=4.
,SpreadSheetXvarFormat=WEEKDATX32.
,GraphAndSpreadSheetTitle=Dow Index By Trading Day
,ForceVzero=YES);

%MACRO Wide_Plots_PROC_GPLOTT_ForAnyData
(Path=
,Data=
,Yvar=
,YvarFormat=
,YvarLabel=
,Xvar=
,XvarFormatForFullRange=
,XvarFormatForSubRange=
,XvarFormatForALTtext=
,XvarLabel=
,CycleVar=
,CycleVarFormat=
,DateRangeFormat=
,LinkableSpreadSheets=NO
,SpreadSheetYvarFormat=
,SpreadSheetXvarFormat=
,GraphAndSpreadSheetTitle=
,ForceVzero=YES
);

%MACRO CreateYearLinks_GPLOTT;
%DO j = 1 %TO &NumberOfYears %BY 1;
  LINK="Year_&&Year&j...html" UNDERLINE=1 "Go To &&Year&j" UNDERLINE=0 " "
%END;
%MEND CreateYearLinks_GPLOTT;
```

```

/* START of SetUp Code not unique to use of SAS/GRAPH PROC GPLOT */

PROC MEANS DATA=&Data MIN MAX NOPRINT;
VAR &Yvar &Xvar;
OUTPUT OUT=WORK.MinMax MIN=MinY StartDate MAX=MaxY EndDate;
RUN;

DATA _NULL_;
SET MinMax;
%if %upcase(&ForceVzero) EQ YES %then %do;
MinY = 0;
%end;
CALL SYMPUT('MinY',TRIM(LEFT(MinY)));
CALL SYMPUT('MaxY',TRIM(LEFT(MaxY)));
CALL SYMPUT('RangeY',TRIM(LEFT(MaxY - MinY)));
CALL SYMPUT('DateRange',
    TRIM(LEFT(PUT(StartDate,&DateRangeFormat))) ||
    ' to ' ||
    TRIM(LEFT(PUT(EndDate,&DateRangeFormat))));
RUN;

PROC SUMMARY DATA=&Data NWAY;
CLASS Year;
VAR &Xvar;
OUTPUT OUT=WORK.EachYearWithStartDateEndDate MIN=StartDate MAX=EndDate;
RUN;

DATA _NULL_;
SET WORK.EachYearWithStartDateEndDate END=LastYear;
CALL SYMPUT('DateRange' || TRIM(LEFT(_N_)),
    TRIM(LEFT(PUT(StartDate,&DateRangeFormat)))
    || ' to ' ||
    TRIM(LEFT(PUT(EndDate,&DateRangeFormat))));
CALL SYMPUT('Year' || TRIM(LEFT(_N_)),TRIM(LEFT(Year)));
IF LastYear;
CALL SYMPUT('NumberOfYears',TRIM(LEFT(_N_)));
RUN;

DATA _NULL_;
SET WORK.EachYearWithStartDateEndDate END=LastYear;
* Want to start on a Monday not later than the first day of data *;
StartDate = MDY(1,1,Year);
DayOfWeekStart = WEEKDAY(StartDate);
IF DayOfWeekStart GT 2
THEN StartDate = StartDate - (DayOfWeekStart - 2);
ELSE
IF DayOfWeekStart EQ 1
THEN StartDate = StartDate - 6;
CALL SYMPUT('StartDateTick' || TRIM(LEFT(_N_)),StartDate);
* Want to end on a Monday not earlier than the last day of data *;
EndDate = MDY(12,31,Year);
DayOfWeekEnd = WEEKDAY(EndDate);
IF DayOfWeekEnd GT 2
THEN EndDate = EndDate + (9 - DayOfWeekEnd);
ELSE
IF DayOfWeekEnd EQ 1
THEN EndDate = EndDate + 1;
CALL SYMPUT('EndDateTick' || TRIM(LEFT(_N_)),EndDate);
RUN;

/* End of SetUp Code not unique to use of SAS/GRAPH PROC GPLOT */

DATA WORK.ToPlot;

```



```

LENGTH HTMLvar $ 512;
SET &Data;
HTMLvar = "alt='"
|| "&XvarLabel: " || TRIM(LEFT(PUT(&Xvar,&XvarFormatForALTtext)))
|| '0D'X /* force a line break in the ALT text display */
|| "&YvarLabel: " || TRIM(LEFT(PUT(&Yvar,&YvarFormat))) || "'";
RUN;

PROC CATALOG CAT=WORK.gseg KILL; RUN; QUIT;

ODS NORESULTS; /* do not open in SAS session */
ODS LISTING CLOSE;

/* START SetUp that applies to all graphs */

ODS GRAPHICS OFF;
GOPTIONS RESET=ALL;
GOPTIONS FTEXT="Arial"; /* for axis values */
GOPTIONS XPIXELS=980 YPIXELS=576;

SYMBOL1 COLOR=blue HEIGHT=0.5 FONT=ZAPF VALUE='E2'X;
SYMBOL2 COLOR=blue HEIGHT=0.5 FONT=ZAPF VALUE='E2'X;
/* Use of FONT=ZAPF VALUE='E2'X results in a smaller HTML file
because many other SAS/GRAPH symbol choices cause the AREA map, used to provide
ALT text, to be generated in a very inefficient manner. When multiple browse
windows are open, not only those browser windows, but also other windows that
the viewing user might have open, can become inoperative. That problem of huge
area maps might have been observed only with very large numbers of plot points
on multi-line plots. Multiple lines multiply the numbers of plots points.
Instead of FONT=ZAPF VALUE='E2'X, an unwise symbol choice for a very dense
multi-line plot would be, e.g., VALUE=DOT. */

AXIS1 LABEL=NONE MAJOR=NONE MINOR=NONE STYLE=0 /* left-hand y axis */
ORDER=&MinY TO &MaxY BY &RangeY
VALUE=(HEIGHT=3PCT);

AXIS2 LABEL=NONE MAJOR=NONE MINOR=NONE STYLE=0 /* right-hand y axis */
ORDER=&MinY TO &MaxY BY &RangeY
VALUE=( HEIGHT=3PCT JUSTIFY=LEFT);

/* END SetUp that applies to all graphs */

AXIS3 LABEL=NONE MAJOR=NONE MINOR=NONE STYLE=0 VALUE=(HEIGHT=2PCT);

TITLE1 FONT='Times New Roman' HEIGHT=4PCT COLOR=black
"&GraphAndSpreadSheetTitle - &DateRange";
TITLE2 FONT='Times New Roman' HEIGHT=4PCT COLOR=CX006600
"Rest mouse on plot points for precise values";
TITLE3 HEIGHT=4PCT FONT='Times New Roman'
COLOR=blue %CreateYearLinks_GPLOT
;
%IF %upcase(&LinkableSpreadSheets) EQ YES %THEN %DO;
TITLE4 HEIGHT=4PCT FONT='Times New Roman' COLOR=blue
LINK="FullRange.XLS" UNDERLINE=1 "Go To Spreadsheet of this data";
%END;

ODS HTML PATH="&Path" (URL=NONE) STYLE=Styles.Minimal GTITLE GFOOTNOTE
BODY="FullRange_DefaultHticks.html" (TITLE="&GraphAndSpreadSheetTitle -
&DateRange");

PROC GPLOT DATA=WORK.ToPlot;
PLOT &Yvar * &Xvar / HTML=HTMLvar NAME="gplot" DESCRIPTION=' '
VAXIS=AXIS1 VZERO CVREF=CXCCCCC LVREF=1 VREF=(&MinY &MaxY)

```

```

        HAXIS=AXIS3 AUTOHREF CHREF=CXCCCCC LHREF=1;
PLOT2 &Yvar * &Xvar /
        VAXIS=AXIS2 VZERO;
FORMAT &Xvar &XvarFormatForFullRange;
RUN; QUIT;

ODS HTML CLOSE;

%IF %upcase(&LinkableSpreadSheets) EQ YES %THEN %DO;

/* Create the companion spreadsheet. */
/* GOPTIONS RESET=ALL;
   Prior GOPTIONS can affect non-SAS/GRAPH PROC output. */
ODS HTML PATH="&Path" (URL=NONE) STYLE=Minimal
        BODY="FullRange.XLS";
/* Use COLSPAN= below so that TITLE lines do not get put only into Column A and
   expand its width to length of longest TITLE line.
   Spread them across sixteen columns, which expand, if needed */
TITLE
JUSTIFY=LEFT "<td COLSPAN=16>&GraphAndSpreadSheetTitle - &DateRange</td>"
JUSTIFY=LEFT
"<td COLSPAN=16>
<a href='\"' \"FullRange_DefaultHticks.html\" '\">
Go To Graph of This Data
</a>
</td>";
PROC PRINT DATA=WORK.ToPlot(DROP=HTMLvar) LABEL NOOBS;
FORMAT &Xvar &SpreadSheetXvarFormat;
FORMAT &Yvar &SpreadSheetYvarFormat;
LABEL &Yvar="&YvarLabel";
RUN;
ODS HTML CLOSE;

%END;

%MACRO LinkedYearlyTrendCharts_AnyData;

%DO i = 1 %TO &NumberOfYears %BY 1;

AXIS3 LABEL=NONE MAJOR=NONE MINOR=NONE STYLE=0 VALUE=(HEIGHT=2PCT ANGLE=-45)
        ORDER = &&StartDateTick&i to &&EndDateTick&i by 7;
TITLE1 FONT='Times New Roman' HEIGHT=4PCT COLOR=black
        "&GraphAndSpreadSheetTitle - &&DateRange&i";
TITLE2 FONT='Times New Roman' HEIGHT=4PCT COLOR=CX006600
        "Rest mouse on plot points for precise values";
TITLE3 HEIGHT=4PCT FONT='Times New Roman' COLOR=blue
        LINK="FullRange_DefaultHticks.html" UNDERLINE=1 "Go To Full Range" UNDERLINE=0 " "
        %DO j = 1 %TO &NumberOfYears %BY 1;
                %IF &&Year&j NE &&Year&i
                        %THEN %DO;
                                LINK="Year_&&Year&j...html" UNDERLINE=1 "Go To &&Year&j" UNDERLINE=0 " "
                                %END;
                        %END;
                ;
        %IF %upcase(&LinkableSpreadSheets) EQ YES %THEN %DO;
TITLE4 HEIGHT=4PCT FONT='Times New Roman' COLOR=blue
        LINK="Year_&&Year&i...XLS" UNDERLINE=1 "Go To SpreadSheet of this data";
        %END;

ODS HTML PATH="&Path" (URL=NONE) STYLE=Minimal GTITLE GFOOTNOTE
        BODY="Year_&&Year&i...html" (TITLE="&GraphAndSpreadSheetTitle - &&DateRange&i");
PROC GPLOT DATA=WORK.ToPlot(WHERE=(Year EQ "&&Year&i"));
PLOT &Yvar * &Xvar / HTML=HTMLvar NAME="gplot&i" DESCRIPTION=' '

```

```

        VAXIS=AXIS1 CVREF=CXCCCCC LVREF=1 VREF=(&MinY &MaxY)
        HAXIS=AXIS3 AUTOHREF CHREF=CXCCCCC LHREF=1;
PLOT2 &Yvar * &Xvar / VAXIS=AXIS2;
FORMAT &Xvar &XvarFormatForSubRange;
RUN; QUIT;
ODS HTML CLOSE;

%IF %upcase(&LinkableSpreadSheets) EQ YES %THEN %DO;

/* Create the companion spreadsheet. */
/* GOPTIONS RESET=ALL;
   Prior GOPTIONS can affect non-SAS/GRAPH PROC output. */
ODS HTML PATH="&Path" (URL=NONE) STYLE=Minimal
   BODY="Year_&&Year&i...XLS";
/* Use COLSPAN= below so that TITLE lines do not get put only into Column A and
   expand its width to length of longest TITLE line.
   Spread them across sixteen columns, which expand, if needed */
TITLE
   JUSTIFY=LEFT "<td COLSPAN=16>&GraphAndSpreadSheetTitle - &&DateRange&i</td>"
   JUSTIFY=LEFT
   "<td COLSPAN=16>
   <a href='\" "Year_&&Year&i...html" '\">
   Go To Graph of This Data
   </a>
   </td>";
PROC PRINT DATA=WORK.ToPlot(WHERE=(Year EQ "&&Year&i") DROP=HTMLvar) LABEL NOOBS;
FORMAT &Xvar &SpreadSheetXvarFormat;
FORMAT &Yvar &SpreadSheetYvarFormat;
LABEL &Yvar="&YvarLabel";
RUN;
ODS HTML CLOSE;

%END;

%END;

%MEND LinkedYearlyTrendCharts_AnyData;

OPTIONS MPRINT;

%LinkedYearlyTrendCharts_AnyData;

ODS LISTING;

%MEND Wide_Plots_PROC_GPLOT_ForAnyData;

```

CONCLUSION

The Tool 1 macro provided here simplifies the task of overcoming a current limitation of the new technology.

The Tool 2 macro enhances what you can get out of ODS Graphics, and is a helpful alternative when the data is such that Tool 1 is not usable.

The third, fourth, and fifth macros, to create subsetting ranked horizontal bar charts, implement a powerful and useful, though simple, concept and graphic method, and these macros work better when built with the new technology.

The Tool 3 macro was really developed as an intermediate step to the Tool 5 macro, but, though you probably will never use Tool 3, it is useful to read the discussion and compare what Tool 3 does as compared to the unsatisfactory results that ODS Graphics would deliver without the intelligence that is built into Tools 3 and 5.

The Tool 4 macro is a niche solution useful if you want to get the maximum information into a slide or a Word page.

The remaining solutions, Tool 6 for ODS Graphics and Tool 7 for SAS/GRAPH, are not general-purpose tools, but show you how to deliver both an overview and by-segment views of long trends, with links between all of the views, and with ALT text so that the viewer can get at the precise y value and exact date for any plot point, rather than try to guess. **Code for the ODS Graphics solution is presented, but its use is not recommended for reasons described above.** The SAS/GRAPH solution is fully packaged as a single invocable macro, to deliver both the plots but also spreadsheets of the input data to the plots, with the plots and spreadsheets hyperlinked back and forth.

The Tool 8 macro generalizes Tool 7 to handle any multi-year daily measurement data set, with macro parameters for variable names, display formats, and labels. Tool 8 could be used as a model to create macros to support other ranges and subranges.

REFERENCES

1. Bessler, LeRoy. "Intelligent Production Graphic Reporting Applications", *Proceedings of the Seventeenth Annual SAS Users Group International Conference*, Cary, NC, USA: SAS Institute Inc., 1992.
<http://www.sascommunity.org/sugi/SUGI92/Sugi-92-34%20Bessler.pdf>
2. Bessler, LeRoy. "Software-Intelligent Application Development: Reliability, Reusability, Extendibility, and Maintainability for Strong Smart Systems in an Ever-Changing World", *Proceedings of the SAS Global Forum 2008 Conference*, Cary, NC, USA: SAS Institute Inc., 2008.
<http://www2.sas.com/proceedings/forum2008/001-2008.pdf>
3. Bessler, LeRoy. "Data Visualization Tips and Techniques for Effective Communication", *Proceedings of the Pharmaceutical SAS Users Group 2013 Conference*, Chicago, IL, USA: 2013. www.PharmaSUG.org

AUTHOR INFORMATION

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

LeRoy Bessler PhD
Bessler Consulting and Research
Converting Complexity Into Clarity™
Strong Smart Systems™
1-262-512-1729
Le_Roy_Bessler@wi.rr.com

Dr. LeRoy Bessler has presented at software user conferences in the US, Canada, and Europe, on topics such as effective visual communication (using graphs, tables, web pages, or color), highly formatted Excel reporting from SAS, custom-developed tools to assist SAS server administrators, users, and managers, and Software-Intelligent Application Development methods to maximize Reliability, Reusability, Maintainability, Extendibility, and Flexibility. His SAS experience includes application development and supporting users, servers, software, and data.

SAS, SAS/GRAPH, 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.

Converting Complexity Into Clarity and Strong Smart Systems are trademarks of LeRoy Bessler PhD.