

Paper 090-29

## Pop-Ups, Drill-Downs, and Animation

Mike Zdeb, University@Albany School of Public Health, Rensselaer, NY

### ABSTRACT

Several features have been added to SAS/GRAPH® that allow a user to go beyond a static display of information in a chart, plot, or map. Many of these features are designed for Web-based displays. They include: pop-ups (boxes populated with additional information that will pop-up when a mouse is dragged over different output areas); drill-downs (output areas, including legends, linked to additional information contained in other web pages); animation (including controls that allow a user to page through an animated display on a frame-by-frame basis or to watch animation at a user-specified time delay between frames). These features of SAS/GRAPH are demonstrated in a series of examples using PROC GCHART, PROC GPLOT, and the JAVAMETA device driver. The examples show how to add pop-ups, drill-downs, and animation to graphical displays of information.

### INTRODUCTION

This paper is an introduction to the creation of graphical presentations of data using SAS/GRAPH, the JAVAMETA device driver, and the Metaview Applet. Use of the JAVAMETA device driver results in the creation of files in metagraphics format. The Metaview Applet generates graphical displays based on the contents of the metagraphics files. What are metagraphics? Rather than explain what they are, a file of metagraphics can be generated quite easily and then viewed.

```
*** EX01 - create a file of metagraphics;
goptions reset=all device=javameta gsfname=xxx;
filename xxx 'z:\gtestit_pic1.txt';
proc gtestit pic=1;
run;
```

After running this job, the LOG contains...

```
NOTE: 206 RECORDS APPENDED TO z:\gtestit_pic1.txt
```

Note the word 'appended'. The JAVAMETA driver opens the destination file for the metagraphics in APPEND mode. Though this is sometimes a feature rather than an inconvenience, a new file can be generated by adding GSFMODE=REPLACE to the GOPTIONS statement. The following are the first five lines of the file GTESTIT\_PIC1.TXT...

```
37 8 106 97 118 97 109 101 116 97 30 0 10 1 13 5
0 0 0 50 8 32 32 32 32 32 32 32 32 51 18 57
46 48 48 46 48 48 77 48 80 48 56 50 49 50 48 48
50 52 0 62 1 63 0 0 64 51 33 65 2409 1785 67 0
57 33 1 35 73 1 9 72 101 108 118 101 116 105 99 97
```

This looks much different than the content of other typical files of graphics: GIF, JPEG, PNG, PDF, postscript. Just as an application is needed to convert the content of one of these 'typical' files into a display of graphics, the Metaview Applet is needed to interpret the above codes and create a display. There are a number of ways to use the Applet. One is to add an HTML statements to the beginning and end of the metagraphics file. At a minimum, this could be...

```
<APPLET CODE="MetaViewApplet.class" CODEBASE="file:///C:\SAS\Shared Files\applets"
ARCHIVE="metafile.zip" Width=800 Height=600>
<PARAM NAME="MetaCodes" VALUE="
37 8 106 97 118 97 109 101 116 97 30 0 10 1 13 5
0 0 0 50 8 32 32 32 32 32 32 32 32 51 18 57
46 48 48 46 48 48 77 48 80 48 56 50 49 50 48 48
...more lines of metagraphics...
6 49 6 4 594 383 1 69 60 7 49 6 4 660 383 1
71 60 8 49 6 4 726 383 1 73 60 9 49 6 4 792
383 1 75 33 32">
</APPLET>
```

The HTML tag APPLETT is used with several attributes: CODE- the applet to run (the SAS-supplied Metaview Applet); CODEBASE-the path to the applet (in this case, a local directory); ARCHIVE-a compressed collection of applet components (the Metaview Applet is stored within the compressed file METAFIILE.ZIP). Two additional attributes are used, WIDTH and HEIGHT, to control the size of the display in pixels (they are not necessary, but the display is quite small without them). The tag PARAM is used to provide details to the Applet as to how it should run. When displayed in a browser, the HTML + metagraphics will produce the following image.



Notice that in addition to the image, the JAVAMETA device driver adds a ZOOM control to the bottom of the image. Subsequent examples will show how to eliminate this control and how to add others.

The output delivery system (ODS) can be used to have SAS/GRAPH write a file of HTML + metagraphics rather than just metagraphics as was done in example EX01.

```
*** EX02 - create a file of HTML + metagraphics;
goptions reset=all device=javameta;
ods listing close;
ods html file='z:\gtestit_pic1.html';
proc gtestit pic=1;
run;
ods html close;
ods listing;
```

## THE DATA - CANADIAN POPULATION

All the remaining examples use the following data set. The data comprise the population of Canada and its provinces from 1951 to 2001 in 5-year intervals (data obtained from Statistics Canada). If you are familiar with Canada, you will notice that there are no data for the Nunavut Territory in 2001. Nunavut, formed out of the eastern part of the Northwest Territory (NWT), became a territory in 1999. The 2001 population of the NWT is that of the Nunavut Territory and the NWT combined. The populations are all in 1000s. For example, the 2001 population of Canada in the data set is 30,007,000.

```
*** EX03 - CANADA AND PROVINCIAL POPULATIONS (IN 1000s);
*** PROVINCE NUMBERS AND NAMES;
proc format;
value $province
'00' = 'CANADA'          '10' = 'NEWFOUNDLAND'      '11' = 'PRINCE EDWARD ISLAND'
'12' = 'NOVA SCOTIA'    '13' = 'NEW BRUNSWICK'    '24' = 'QUEBEC'
'35' = 'ONTARIO'        '46' = 'MANITOBA'         '47' = 'SASKATCHEWAN'
'48' = 'ALBERTA'        '59' = 'BRITISH COLUMBIA' '60' = 'YUKON'
'61' = 'NORTHWEST TERRITORIES'
;
run;
```

```

*** POPULATIONS (1951-2001);
data canada_pop;
infile datalines dsd;
input province : $2. p1951 p1956 p1961 p1966 p1971 p1976 p1981 p1986 p1991 p1996 p2001;
datalines;
00,14009,16081,18238,20015,21568,22993,24343,25309,27297,28847,30007
10,361,415,458,493,522,558,568,568,568,552,513
11,98,99,105,109,112,118,123,127,130,135,135
12,643,695,737,756,789,829,847,873,900,909,908
13,516,555,598,617,635,677,696,709,724,738,729
24,4056,4628,5259,5781,6028,6234,6438,6532,6896,7139,7237
35,4598,5405,6236,6961,7703,8264,8625,9102,10085,10754,11410
46,777,850,922,963,988,1022,1026,1063,1092,1114,1120
47,832,881,925,955,926,921,968,1010,989,990,979
48,940,1123,1332,1463,1628,1838,2238,2366,2546,2697,2975
59,1165,1398,1629,1874,2185,2467,2744,2883,3282,3725,3908
60,9,12,15,14,18,22,23,24,28,31,29
61,16,19,23,29,35,43,46,52,57,65,64
;
run;

```

## POP-UPS

The ACTIVEX, JAVA, and JAVAMETA device drivers all can be used to add pop-ups to graphical displays. However, when using either ACTIVEX or JAVA, the content of the pop-up box can be altered in only very limited ways. The Metaview Applet tag TIP= tag allows a great degree of control over both the content of a pop-up box and the manner in which the pop-up box is associated with an area of the graphics display. Also, a TIP (pop-up) need not be associated with each area of a graphics display.

```

*** EX04 - ADD POP-UP BOXES WITH PROVINCIAL POPULATIONS IN 10-YEAR INTERVALS;
data pop_tips;
set canada_pop;
*** ADD A NEW VARIABLE (POPUP) WITH A TIP= TAG;
popup = 'tip=[' ||
        quote('POPULATION (1000s)') ||
        quote('2001: ' || put(p2001,comma6.)) ||
        quote('1991: ' || put(p1991,comma6.)) ||
        quote('1981: ' || put(p1981,comma6.)) ||
        quote('1971: ' || put(p1971,comma6.)) ||
        quote('1961: ' || put(p1961,comma6.)) ||
        quote('1951: ' || put(p1951,comma6.)) ||
        ']';
run;

*** SELECT THE JAVAMETA DEVICE DRIVER, SET SIZE, FONT;
goptions reset=all
device=javameta
xpixels=800
ypixels=600
ftext='HelveticaBold'
htext=3
gunit=pct;

*** DEFINE PATTERN FOR BARS, AXES DEFINITIONS;
pattern1 v=s c=green;

axis1
label=('POPULATION (THOUSANDS)')
order=(0 to 12000 by 2000);

axis2
label=('PROVINCE')
value=(j=r)
order=('NEWFOUNDLAND' 'PRINCE EDWARD ISLAND' 'NOVA SCOTIA' 'NEW BRUNSWICK' 'QUEBEC' 'ONTARIO'
'MANITOBA' 'SASKATCHEWAN' 'ALBERTA' 'BRITISH COLUMBIA' 'YUKON' 'NORTHWEST TERRITORIES');

title1 h=5 "CANADIAN PROVINCES - 2001 POPULATION";

*** USE ODS, DEFINE PARAMETERS FOR METAVIEW APPLLET;
ods listing close;
ods html file='z:\canada_p2001.html'
        parameters=("ZoomControlEnabled"="false"
                    "DataTipStyle"="stick_fixed");

*** CREATE A 3-DIMENSIONAL HORIZONTAL BAR CHART;
*** USE THE HTML OPTION TO ADD POP-UP BOXES TO THE DISPLAY;
proc gchart data=pop_tips;
hbar3d province/
sumvar = p2001

```

```

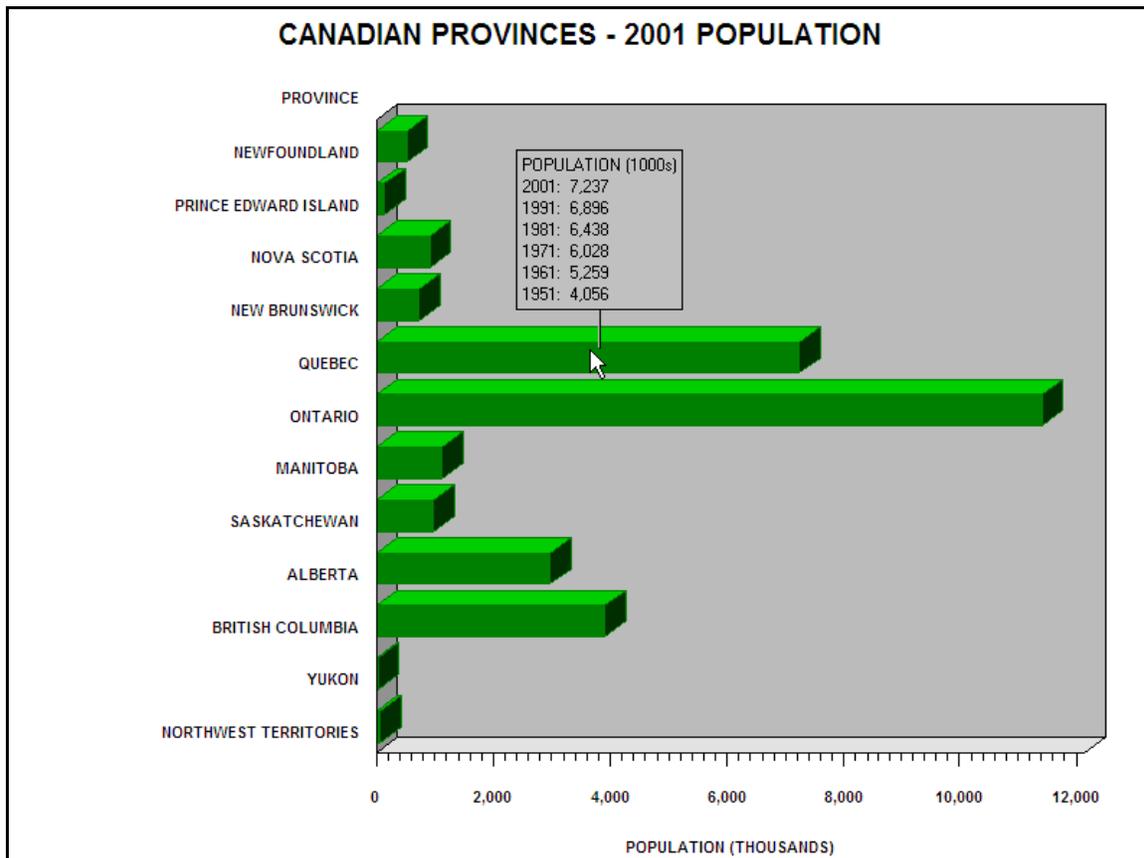
shape=block
raxis=axis1
maxis=axis2
nostats
cframe=graybb
html=popup
;
where province ne '00';
format province $province. p2001 comma.;
run;
quit;

ods html close;
ods listing;

```

The above example produces the following image when displayed in a browser. The mouse is positioned over the horizontal bar for Quebec and the pop-up box shows the population of the province in 10-year intervals from 1951 to 2001. The pop-up box is associated to the horizontal bar with a stick.

The contents of the pop-up box is set in the first data step of EX04. The data set containing the provincial populations is read.



The variable POPUP comprises a combination the TIP= tag, text, and values of the variables containing population. A TIP= tag always begins with 'TIP=' and ends with ']'. Between the brackets are a series of quoted strings. Each new quoted string produces a new line in the pop-up box (it is very easy to populate the pop-up box with values of variables in the data set).

The JAVAMETA device driver is selected in the GOPTIONS statement. Most of the remaining SAS code is similar to what is used to produce a horizontal bar chart without any extra features. The differences are seen in the ODS HTML statement and in the HBAR3D statement in PROC GCHART. The ODS HTML statement does a number of things: specifies the file to be created; uses a PARAMETERS= option to turn off the default ZOOMCONTROLENABLED (shown in the output produced by EX01) and to set the DATATIPSTYLE as a stick fixed to points within the graph area (other styles are HIGHLIGHT and

STICK). The HTML= option in PROC GCHART specifies the variable (POPUP) containing additional information used when the chart is displayed in a browser.

## DRILL-DOWNS

Links to other information can be added to output produced with the JAVAMETA device driver by using either the HREF= or MENU= tag. The HREF= tag places a link graph areas. When the graphics output is displayed in a browser and a mouse is clicked in a linked area, the browser will follow the link and open the linked file. The HREF= tag is limited to one link per area. The MENU= tag defines a menu that appears in a pop-up box. Items in the menu can be linked to other files and more than one item can appear in the menu allowing more than one link per area. The next example links each bar in the chart to a plot of population in 5-year intervals from 1951 to 2001. The SAS code that produces the plots is shown in the appendix and uses the data set created in EX03. The JAVAMETA device driver is used produce the plots and each symbol in the plots is associated with a TIP= tag, a pop-up box that shows a population.

```

*** EX05 - ADD A LINK TO EACH BAR IN THE HORIZONTAL BAR CHART;
data pop_tips;
set canada_pop;
poplink = 'ALT="CLICK FOR PLOT" href=' || quote('plotprov' || province || '.html');
run;

goptions reset=all
device=javameta
xpixels=800
ypixels=600
ftext='HelveticaItalicBold'
htext=3
gunit=pct
;

pattern1 v=s c=green;

axis1
label=('POPULATION (THOUSANDS)')
order=(0 to 12000 by 2000)
;

axis2
label=('PROVINCE')
value=(j=r)
order=('NEWFOUNDLAND' 'PRINCE EDWARD ISLAND' 'NOVA SCOTIA' 'NEW BRUNSWICK' 'QUEBEC' 'ONTARIO'
'MANITOBA' 'SASKATCHEWAN' 'ALBERTA' 'BRITISH COLUMBIA' 'YUKON' 'NORTHWEST TERRITORIES')
;

title1 h=5 "CANADIAN PROVINCES - 2001 POPULATION";

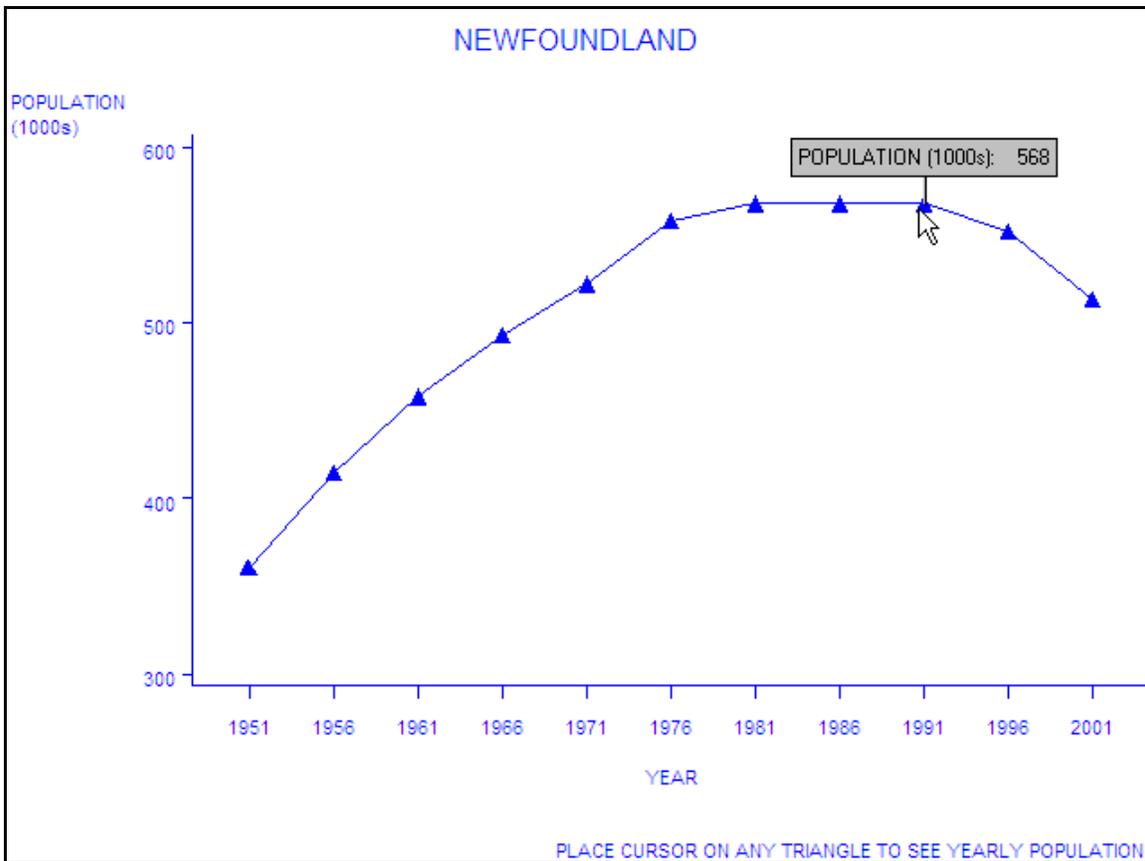
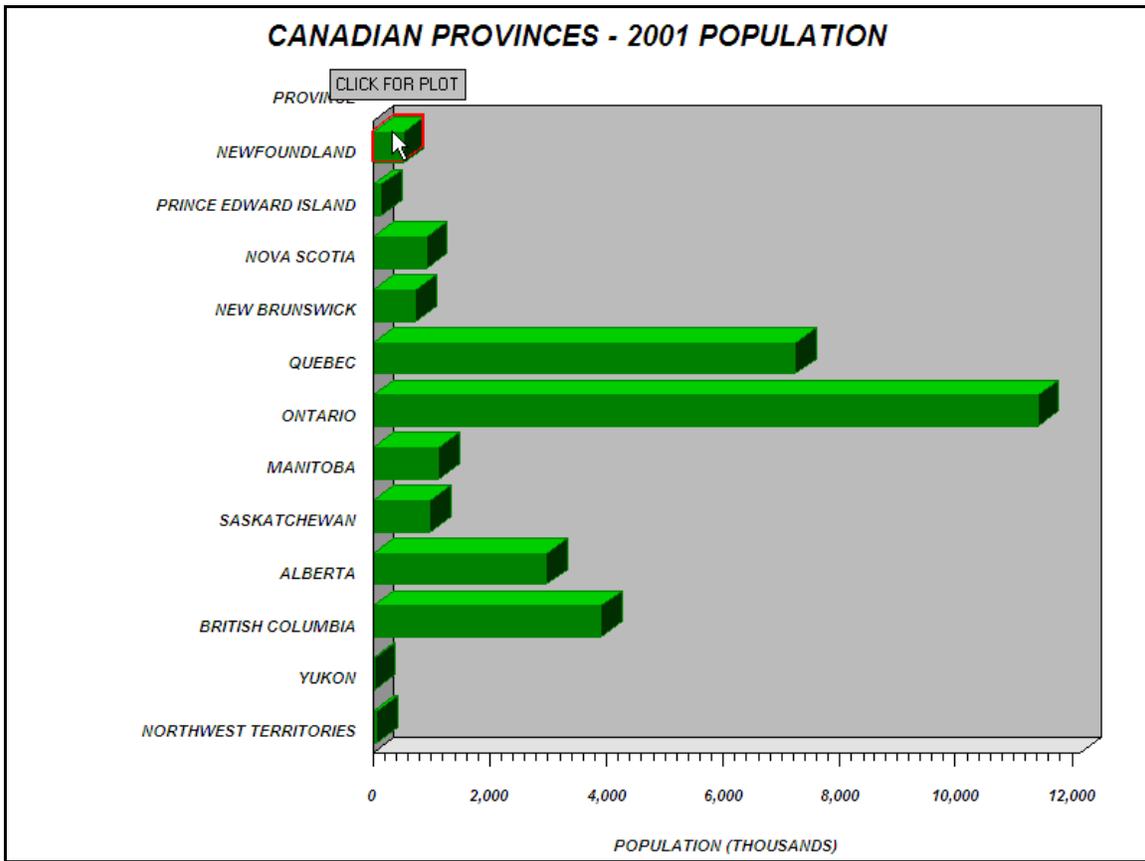
ods listing close;
ods html file='z:\canada_p2001.html' parameters=("ZoomControlEnabled"="false");

proc gchart data=pop_tips;
hbar3d province/
sumvar = p2001
shape=block
raxis=axis1
maxis=axis2
nostats
cframe=graybb
html=poplink
;
where province ne '00';
format province $province. p2001 comma.;
run;
quit;

ods html close;
ods listing;

```

The first data step adds an HREF= tag to the data set. and the HTML= option in PROC GCHART links the tag to the screen display. This tag has both an ALT tag with a message that shows on the screen as the mouse passes over population bar and a link to a plot of populations. The next figure shows what happens when a mouse cursor placed on a bar. Notice that the bar is highlighted in red and the text placed in the ALT= tag within the HREF= tag shows on the screen. If the left mouse button is clicked, a plot is opened in another browser window. All the plots have pop-up boxes showing yearly populations and the plot for Newfoundland looks as shown on the next page.



## ANIMATION

The GIFANIM device driver can be used to create a graphics display as an animate GIF file. When the animated GIF file is created, two options can be specified in a GOPTIONS statement: DELAY=, controlling the time-delay between the various frames within the animated GIF; ITERATION=, controlling the number of times that the loop through the animated GIF is repeated. These options are set at the time of file creation and cannot be changed.

The JAVAMETA device driver can be used to create an animated display of graphics. The file that is created has controls displayed on the screen that: allow a viewer of the display to control the time-delay between frames within the animation; page through the various frames manually, either forward or backward. The next example produces an animated display with on-screen controls. The data set used in this example contains Canadian populations from 1921 through 2001, and population projections for 2001 and 2021 (data obtained from Statistics Canada). Within each year, the data are classified by age group and gender. The first four observations in the data set look as follows...

YEAR	AGE GROUP	POPULATION (1000s)	GENDER
1921	1	-11.3	f
1921	1	9.5	m
1921	2	-55.4	f
1921	2	53.9	m
1921	3	-138.6	f
1921	3	151.6	m
1921	4	-245.7	f
1921	4	275.4	m
1921	5	-365.9	f
1921	5	433.3	m

Notice that female populations (gender = f) are negative. They will appear on the bar chart produced by PROC GCHART output to the right of the y-axis, while male population (gender=m) will be on the left. The age limits for each age group are shown in PROC FORMAT. The data step that creates the data set used in this example is shown in the appendix, as is a portion of the data file used to create the data set.

```
EX06 - CREATE AND ANIMATED GRAPHICS DISPLAY WITH ON-SCREEN CONTROLS;
*** formats used to display age group limits, display negative female population as positive;
proc format;
value age
11 = '0-4'      10 = '5-9'      9 = '10-14'    8 = '15-24'
 7 = '25-34'    6 = '35-44'    5 = '45-54'    4 = '55-64'
 3 = '65-74'    2 = '75-84'    1 = '85+'
;
picture posval low-high='000,009';
run;

*** create an annotate data set to place text on population pyramid;
*** add gender and age-group labels;
data age_labels;
length text $10 style $15;
retain xsys ysys '2' hsys '3' size 3 position '5' x 0
      function 'label' color 'blue' style "Helvetica" when 'a';
do midpoint = 1 to 11;
  text = put(midpoint,age.);
  output;
end;
midpoint=1; ysys='5'; y=100; size=4; style="HelveticaBold";
x=-3000; text='FEMALES' ; position='6'; output;
x= 3000; text='MALES' ; position='4'; output;
x=0 ; text='AGE GROUP'; position='5'; output;
run;

*** choose the JAVAMETA device driver;
goptions reset=all device=javameta
      ftext='Helvetica' htext=3 gunit=pct
      xpixels=800 ypixels=600;

*** use the WEBOUT filename (the JAVAMETA device driver uses the as a default destination);
filename _webout "z:\pyramid.html";

*** set axis and pattern characteristics;
axis1 style=0 value=none label=none color='blue';
axis2 order=(-3000 to 3000 by 1000) label=('POPULATION (1000s)') color='blue';

pattern1 v=s c=cxbdd7e7;
pattern2 v=s c=cx6baed6;
```

```

*** write records to an HTML file - text to precede metagraphics produced by PROC GCHART;
data _null_;
  file _webout;
  input;
  put _infile_;
  datalines;
  <html>
  <title>POPULATION PYRAMIDS - CANADA 1921-2021</title>
  <APPLET CODE="MetaViewApplet.class" CODEBASE="file:///C:\SAS\Shared Files\applets"
    ARCHIVE="metafile.zip" Width=800 Height=600>
  <param name="ZoomControlEnabled" value="false">
  <param name="Metacodes" value="
  ;
run;

*** use a macro and PROC GCHART to produce nine population pyramids;
%macro pyramid;
%do year = 1921 %to 2021 %by 10;

data _null_;
set c1921_2021 end=last;
where year eq &year and age eq 12;
tpop + (abs(pop) * 1000);
if last then call symput('pop',put(tpop,comma12.));
run;

title1 h=4 f='HelveticaBold' c=blue "CANADIAN POPULATION &YEAR - %left(%bquote(&pop))";
title2 h=4 ' ';
proc gchart data=c1921_2021;
where year eq &year and age ne 12;
hbar age /
discrete
sumvar = pop
subgroup = gender
noframe
nostats
space=0
maxis=axis1
raxis=axis2
outline=blue
annotate=age_labels
nolegend
;
format pop posval.;
run;
quit;

%end;
%mend;

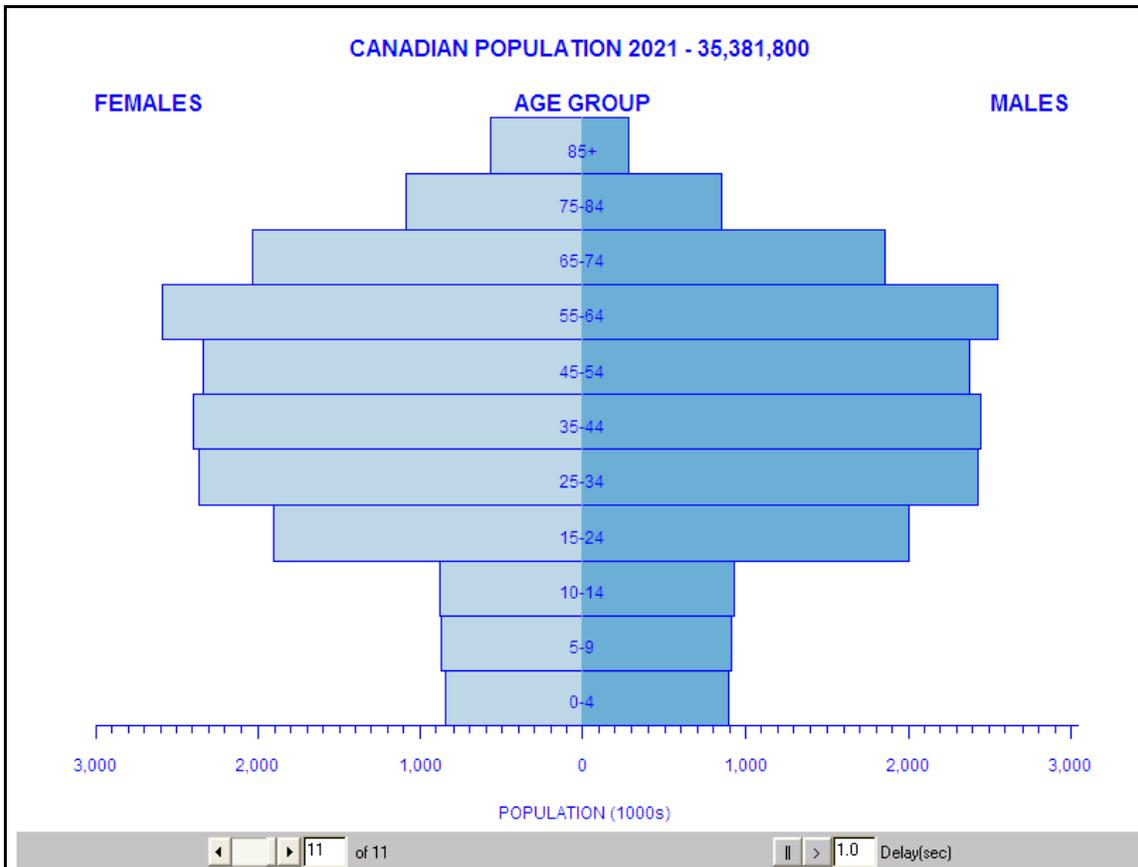
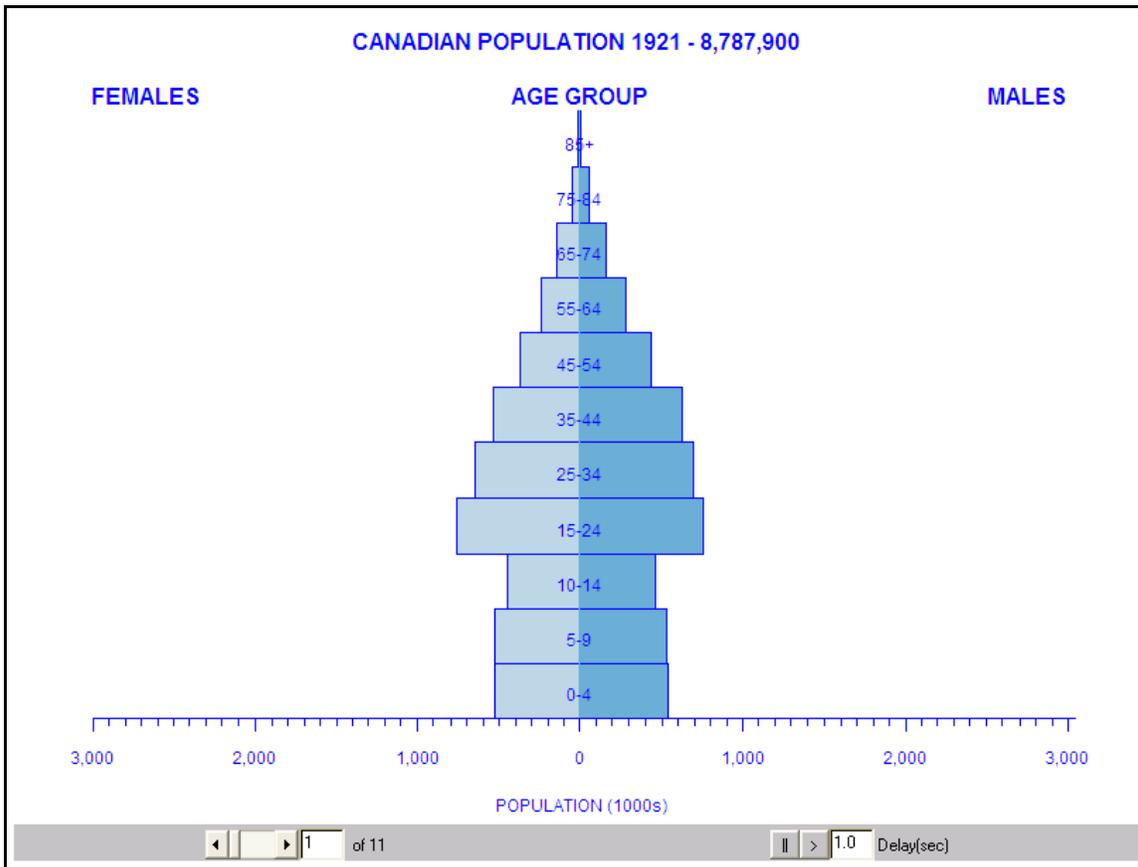
%pyramid;

*** add records to the end of the HTML file;
data _null_;
  file _webout mod;
  input;
  put _infile_;
  datalines;
  ">
  </applet>
  </html>
  ;
run;

*** deassign the _WEBOUT filename;
filename _webout;

```

As in example EX01, ODS is NOT used. Therefore, the file of metagraphics produced with the JAVAMETA device driver only contains metagraphics, no HTML. To display the graphics, HTML statements are needed both preceding and following the metagraphics. A data step is used to write HTML statements to a file prior to using a macro and PROC GCHART to produce the nine population pyramids. The filename `_WEBOUT` is used. One applet parameter is specified, removing the default zoom control. The JAVAMETA device driver uses the file associated with the filename `_WEBOUT` as a default destination and the default mode of writing to that file is APPEND. PROC GCHART appends metagraphics to the first set of HTML statements. After the macro is completed, another data step is used to write the remaining HTML statements needed to display the population pyramids. The last data step uses the MOD option on the FILE statement. The first and last pyramids are shown on the next page. Notice the on-screen controls: the lower-left control enables manual paging through the displays; the lower-right control starts/stops the animation, and the number of seconds in the box (delay between displays) can be changed when the file is viewed.



## LOCAL VERSUS WEB DISPLAY

All the examples thus far have been displayed locally. The SAS code wrote files to a local directory (a directory on the same computer where SAS was run). The HTML statements that precede the metagraphics in each file refer to a local directory as the location of the METAVIEW APPLET, the applet that interprets the metagraphics and creates the on-screen display. For example, the first portion of the HTML file that produce the population pyramids looks as follows...

```
<html>
<title>POPULATION PYRAMIDS - CANADA 1921-2021</title>
<APPLET CODE="MetaViewApplet.class" CODEBASE="file:///C:\SAS\Shared Files\applets"
  ARCHIVE="metafile.zip" Width=800 Height=600>
<param name="ZoomControlEnabled" value="false">
<param name="Metacodes" value="
  37 8 106 97 118 97 109 101 116 97 30 0 10 1 13 5
  0 0 0 50 8 32 32 32 32 32 32 51 18 57
  46 48 48 46 48 48 77 48 80 48 56 50 49 50 48 48
  50 52 0 62 1 63 0 0 64 17 10 65 800 600 67 0
  57 33 1 35 73 1 9 72 101 108 118 101 116 105 99 97
```

The CODEBASE= attribute shows the location of the file METAFILE.ZIP. If ODS is used to create the HTML statements that precede the metagraphics (examples EX02, EX04, EX05), the value of the option APPLETLOC is written to the output file as the location of METAFILE.ZIP. This can be seen by using...

```
proc options option=appletloc;
run;
```

The following is written to the SAS LOG...

```
APPLETLOC=C:\SAS\Shared Files\applets
      Location of Java applets
```

If metagraphics are to be displayed on the web, the value of the applet attribute CODEBASE= must be changed. This can be done by specifying an ODS attribute. Example EX07 is similar to EX02. The only difference is that a new location is specified for METAFILE.ZIP.

```
*** EX07 - create a file of metagraphics for WEB display;
goptions reset=all device=javameta;

ods listing close;
ods html file='z:\gtestit_pic1.html'
      attributes=(codebase="http://www.albany.edu/~msz03/test/sasappl");

proc gtestit pic=1;
run;

ods html close;
ods listing;
```

A portion of the file produced by example EX07 looks as follows...

```
<APPLET NAME="GIDX"
codebase="http://www.albany.edu/~msz03/test/sasappl"
CODE="MetaViewApplet.class"
ARCHIVE="metafile.zip"
MAYSCRIPT
Width=640
Height=480
Align=TOP
>
```

The value of the CODEBASE= attribute is a location on a web server (specifically, web space managed by the author on the U@Albany web site). Both the CODE= and ARCHIVE= attributes have the same values as used for local display. The MAYSCRIPT attribute indicates that the applet is allowed to access the scripting objects of the web page. The default size (in pixels) of the graphic is 640x480. These can be changed using the XPIXELS= and YPIXELS= options within a GOPTIONS statement.

## DROP-DOWN MENU FOR WEB DISPLAY

Another feature of using the JAVAMETA device driver and Metaview Applet is the ability to keep HTML statements and metagraphics codes in separate files. Statements within a file of HTML statements point to metagraphics stored in separate files. When the HTML file is displayed in a browser, a drop-down menu appears at the bottom of the display. Items in the menu are linked to metagraphics. One advantage of this method is that individual files linked to menu items can be

updated/replaced without having to regenerate all the graphics associated with the display. When the metagraphics are embedded in the file of HTML statements (as in the population pyramid example), the entire file must be replaced even if a change in only one year is made. The SAS code is similar to that found in example EX04 in which a horizontal bar chart is created using data from only one year, 2001. PROC GCHART is run within a macro that produces eleven separate files of metagraphics, population displays from 1951 to 2001 in 5-year intervals. ODS is not used. The fileref \_WEBOUT is used. As described in the last example, it is the default destination for output generated with the JAVAMETA device driver. The physical file name for the metagraphics changes with every iteration of the macro.

```

*** EX08 - CREATE FILES OF METAGRAPHICS;
*** MENU-SELECTED CHARTS OF POPULATION, 1951 TO 2001 IN 5-YEAR INTERVALS ;
goptions reset=all device=javameta
        ftext='HelveticaBold' htext=3 gunit=pct
        xpixels=800 ypixels=600;

pattern1 v=s c=green;

axis1 label=('POPULATION (THOUSANDS)') order=(0 to 12000 by 2000);

axis2 label=('PROVINCE') value=(j=r)
order=('NEWFOUNDLAND' 'PRINCE EDWARD ISLAND' 'NOVA SCOTIA' 'NEW BRUNSWICK' 'QUEBEC' 'ONTARIO'
      'MANITOBA' 'SASKATCHEWAN' 'ALBERTA' 'BRITISH COLUMBIA' 'YUKON' 'NORTHWEST TERRITORIES');

%macro popbars;
%do year=1951 %to 2001 %by 5;

data _null_;
set canada_pop;
where province eq '00';
pop = 1000 * p&year;
call symput('pop',put(pop,comma15.));
run;

filename _webout "z:\popbar&year..txt";

title1 h=5 "CANADIAN POPULATION &year (%left(%bquote(&pop)))";
proc gchart data=canada_pop;
hbar3d province/
sumvar = p&year
shape=block
raxis=axis1
maxis=axis2
nostats
cframe=graybb
;
where province ne '00';
format province $province. p&year comma.;
run;
quit;

filename _webout;
%end;
%mend;

%popbars;

```

A file of HTML statements is needed to display the contents of the eleven files of metagraphics. The first portion of the file looks similar to that described earlier for web display of metagraphics. There are new parameter statements in the file: a series that points to the location of the eleven files of meta graphics; a series that defines the text that appears in a drop-down menu.

```

<html>
<head>
<title>CANADIAN PROVINCIAL POPULATION (SOURCE: STATISTICS CANADA)</title>
</head>
<applet
  archive="http://www.albany.edu/~msz03/test/sasappl/metafile.zip"
  code="MetaViewApplet.class" mayscript width="800" height="600" align="CENTER">

  <param name="ZoomControlEnabled" value="false">

  <param name="Metacodes" value="http://www.albany.edu/~msz03/test/popbar2001.txt">
  <param name="Metacodes1" value="http://www.albany.edu/~msz03/test/popbar1996.txt">
  <param name="Metacodes2" value="http://www.albany.edu/~msz03/test/popbar1991.txt">
  <param name="Metacodes3" value="http://www.albany.edu/~msz03/test/popbar1986.txt">
  <param name="Metacodes4" value="http://www.albany.edu/~msz03/test/popbar1981.txt">

```

```

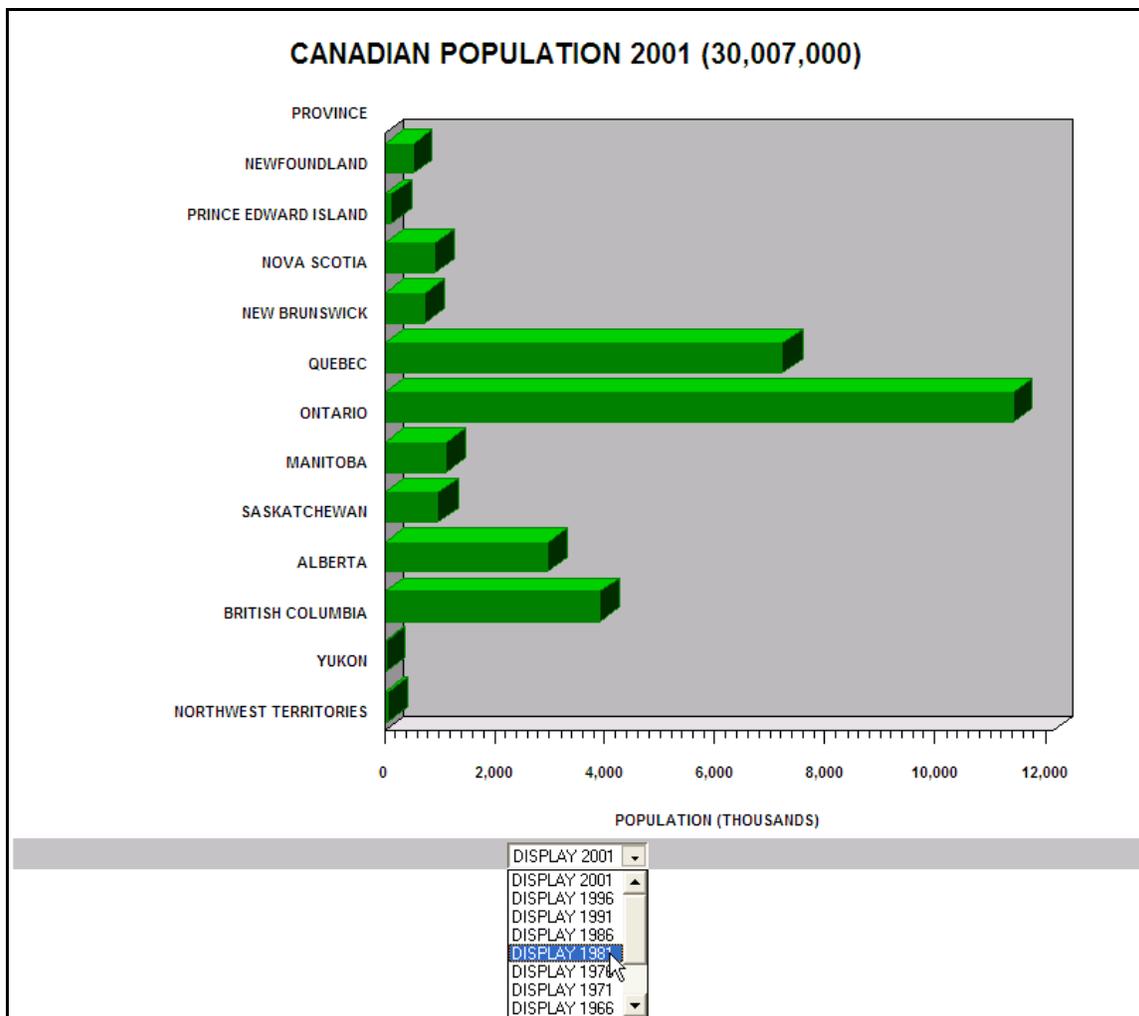
<param name="Metacodes5" value="http://www.albany.edu/~msz03/test/popbar1976.txt">
<param name="Metacodes6" value="http://www.albany.edu/~msz03/test/popbar1971.txt">
<param name="Metacodes7" value="http://www.albany.edu/~msz03/test/popbar1966.txt">
<param name="Metacodes8" value="http://www.albany.edu/~msz03/test/popbar1961.txt">
<param name="Metacodes9" value="http://www.albany.edu/~msz03/test/popbar1956.txt">
<param name="Metacodes10" value="http://www.albany.edu/~msz03/test/popbar1951.txt">

<param name="MetacodesLabel" value=" DISPLAY 2001 ">
<param name="Metacodes1Label" value=" DISPLAY 1996" >
<param name="Metacodes2Label" value=" DISPLAY 1991" >
<param name="Metacodes3Label" value=" DISPLAY 1986" >
<param name="Metacodes4Label" value=" DISPLAY 1981" >
<param name="Metacodes5Label" value=" DISPLAY 1976" >
<param name="Metacodes6Label" value=" DISPLAY 1971" >
<param name="Metacodes7Label" value=" DISPLAY 1966" >
<param name="Metacodes8Label" value=" DISPLAY 1961" >
<param name="Metacodes9Label" value=" DISPLAY 1956" >
<param name="Metacodes10Label" value=" DISPLAY 1951" >

</applet>
</html>

```

The VALUE= attribute of the METACODES= parameter is points to the location of the metagraphics (a directory on a web server). The VALUE= attribute of the METACODESLABEL= parameter describes the text that appears in a drop down menu below the bar chart. The first file in the list contains data from 2001 and it is the chart that appears first when the file of HTML commands is opened. The following figure shows the screen display when the drop-down menu is used to select a chart for another year. The default zoom control is removed by setting the value ZOOMCONTROLENABLED parameter to FALSE.



## LOOSE ENDS

Not all the possible uses of the JAVAMETA device driver and the Metaview Applet have been described. For example, it is possible to use both the TIP= and HREF= tags in one display. That combination would add both pop-up boxes and a drill-downs to graphics areas. There is also a MENU= tag that will produce a pop-up box with links to other files, a combination pop-up and drill-down. The MENU= tag allows a graphics area to be linked to multiple destinations.

The TIP= tag results in a pop-up box. In addition to the STICK\_FIXED value assigned to the DATATIPSTYLE parameter in example EX04, there are two other alternatives, HIGHLIGHT and STICK. The HIGHLIGHT value puts a colored border around the graphics area associated with the pop-up box. The default border color is red, but that can be changed by changing the value of the DATATIPHIGHLIGHTCOLOR parameter. The STICK value allows the stick connecting the pop-up box to move within a graphics area rather than being fixed as is the behavior with the STICK\_FIXED value.

The HelveticaBold font was used on most examples. There are eight fonts available for use with the JAVAMETA device driver: TimesRoman, Courier, Dialog, DialogInput, ZapfDingbats, serif, sansserif, monospaced. All these fonts except for ZapfDingbats allow the following suffixes: Bold, Italic, ItalicBold. The ZapfDingbats font allows for the addition of a variety of characters to a graphics display. To see some the regular characters associated with the special ZapfDingbats characters, the following can be used.

```
options reset=all device=javameta htext=10 gunit=pct;

title1 f='Helvetica' 'ABCDEFGHIJKLMNOPQRSTUVWXYZ';
title2 f='ZapfDingbats' 'ABCDEFGHIJKLMNOPQRSTUVWXYZ';
title3 ' ';
title4 f='Helvetica' 'abcdefghijklmnopqrstuvwxyz';
title5 f='ZapfDingbats' 'abcdefghijklmnopqrstuvwxyz';
title6 ' ';
title7 f='Helvetica' '1234567890!@#$$%^&*()_';
title8 f='ZapfDingbats' '1234567890!@#$$%^&*()_';

ods listing close;
ods html file='z:\helv_zapf.html';

proc gslide;
run;
quit;

ods html close;
ods listing;
```

The output produced by this code is shown on the next page. Notice that the zoom control is used to reduce the size of the characters to 75% of normal size.

One other item that has not been mentioned is the META2HTM macro available in Version 9. It is designed to assist in the generation of web pages that use the Metaview Applet.

## RESOURCES

The on-line documentation in Version 9 for Windows contains extensive material on the Metaview Applet. Most of the documentation is contained in two sections: Developing Web Presentations for the Metaview Applet, Enhancing Presentations for the Metaview Applet. The documentation is also available on the web at...

<http://v9doc.sas.com/sasdoc/>

and

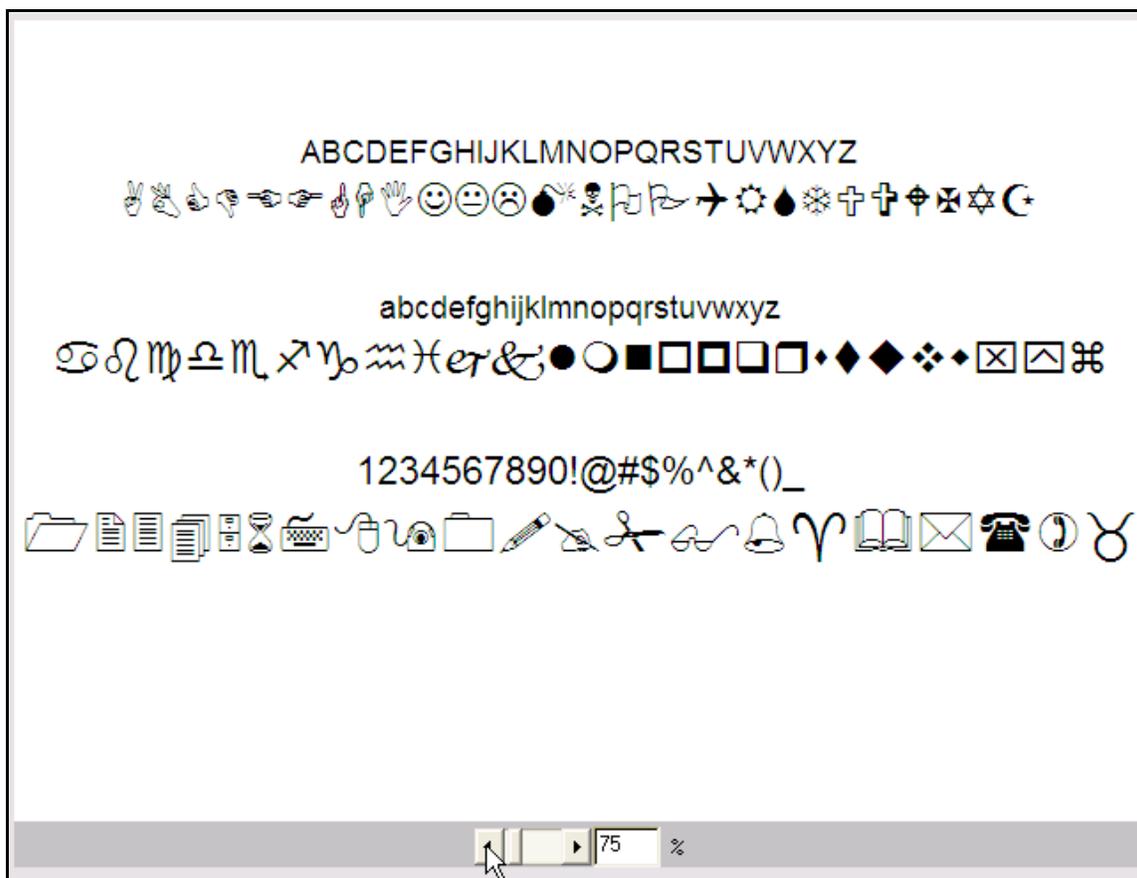
<http://support.sas.com/91doc/docMainpage.jsp>

The web-based documentation for Version 8 contains no mention of JAVAMETA or the Metaview Applet. However, the following link does have information for Version 8 users...

<http://support.sas.com/rnd/datavisualization/webgraphs/v8v81Usage/metaviewapplet.htm>

SAS code shown at the above URL was very helpful in developing this presentation.

A search through the on-line SUGI Proceedings found no papers that had uses of the Metaview Applet as their principle topic. There are a number of papers, however, on the topic of web-based graphics, including, but not limited to the following...

**SUGI 28**

Paper 14-28

Web Enabling Your Graphs With HTML, ActiveX, and Java Using SAS/Graph and the Output Delivery System

Curtis Smith

Paper 84-28: A Drill-Down Diet: An Example of a Dynamic Detail Access on the Web Using SAS/GRAPH and ODS

Nadia Redmond

Paper 127-28: Dynamic Behavior from Static Web Applications

Ted Durie

Paper 149-28: Creating Drill-Down Graphs Using SAS/Graph and the Output Delivery System

Curtis Smith

Paper 207-28: What's In a Map: A Macro-Driven Drill-Down Geo-Graphical Representational System

Louise Hadden

**SUGI-27**

Paper 137-27: Enhancements to SAS/GRAPH in V9

Himesh Patel, Ravi Devarajan

Paper 177-26: The New User Interface of V9 Graphing Tools: A Usability Case Study

Frank Wimmer

Paper 220-27: Show Your Graphs and Tables at Their Best on the Web with ODS

Francesca Pierri, LeRoy Bessler

## CONCLUSION

The JAVAMETA device driver and the Metaview Applet make it quite easy to develop dynamic, web-based presentations of information in graphical format. ODS can be used to automatically write the HTML statements necessary to display the metagraphics produced by the JAVAMETA device driver. The various tags, parameters, and values available within the Metaview Applet enable: pop-up boxes (and pop-up menus); drill downs (and drop-down menus); animation.

## CONTACT INFORMATION

The author can be contacted using [e-mail...msz03@albany.edu](mailto:msz03@albany.edu)

by phone...518-402-6479

or by regular mail...

Mike Zdeb  
 U@Albany School of Public Health  
 1 University Place  
 Rensselaer, NY 12144-3456

## APPENDIX

### Example EX05

In example EX05, plots produced using PROC GPLOT are linked to the various bars in a horizontal bar chart. Also, each triangle in each plot is associated with a pop-up box showing the population in the given year. The SAS code used to produce the plots is as follows...

```

*** produce plots linked to bars in charts produced in example EX05;

*** transpose the data set of provincial populations;
proc transpose data=canada_pop out=tcanada_pop (rename=(col1=pop));
var p1951 p1956 p1961 p1966 p1971 p1976 p1981 p1986 p1991 p1996 p2001;
by province;
run;

*** add the variable year to the transposed data set;
data tcanada_pop;
set tcanada_pop;
year = input(substr(_name_,2,4),4.);
drop _name_;
run;

*** set graphics options (specify the JAVAMETA device driver);
goptions reset=all device=javameta
        ftext='Helvetica' htext=3 gunit=pct
        colors=(blue);

*** choose a symbol and axis characteristics;
symbol1 h=2 f=marker v='C' i=join;

axis1 label=('YEAR') order=1951 to 2001 by 5 offset=(5,5)pct minor=none;
axis2 label=(j=1 'POPULATION' j=1 '(1000s)') minor=none;

footnote j=r 'PLACE CURSOR ON ANY TRIANGLE TO SEE YEARLY POPULATION ';

*** a macro that creates one plot for Canada and each province;
%macro plots;
*** choose the data for each plot (11 observations for Canada and each province);
%do firstobs=1 %to 133 %by 11;

data temp;
set tcanada_pop (firstobs=&firstobs obs=%eval(&firstobs+10)) end=last;
popup = 'tip=[ ' || quote('POPULATION (1000s): ' || put(pop,comma.)) || ']';
*** create two macro variables, one is the province number, the other is the province name;
if last then do;
    call symput('provnum',put(province,2.));
    call symput('provname',put(province,$province.));
end;
drop province;
run;

ods html file="z:\provplot&provnum.html"
        parameters=("ZoomControlEnabled"="false" "DataTipStyle"="stick_fixed");

```

```

title h=5 "%trim(&provname)";
proc gplot data=temp;
plot pop*year / haxis=axis1 vaxis=axis2 noframe html=popup;
format pop comma.;
run;
quit;

ods html close;

%end;
%mend;

*** use the macro to create the plots;
ods listing close;
%plots;
ods listing;

```

### Example EX06

The data set used in example EX06 was produced using the following data step...

```

*** produce the data set used in example EX06;

data c1921_2021 (keep=year age gender pop);
infile "f:\sugi29\data\c1921_2021.txt" end=last;
input year age t m f;
age = 12 - age;
pop = m/1000; gender = 'm'; output;
pop = -f/1000; gender = 'f'; output;
label
age = 'AGE GROUP'
pop = 'POPULATION (1000s)'
;
run;

```

The data referred to in the INFILE statement contains Canadian populations from 1921 to 2001 in 10-year intervals, with twelve records per year. Each record contains: year, age group, total population, male population, female population. The age groups are described in example EX06. Data for the first year and last year look as follows...

1921 00	8787900	4529600	4258300
1921 01	1058900	534100	524800
1921 02	1050200	529400	520700
1921 03	914300	461900	452400
1921 04	1518000	756900	761100
1921 05	1343300	693000	650300
1921 06	1162800	630600	532100
1921 07	799200	433300	365900
1921 08	521100	275400	245700
1921 09	290200	151600	138600
1921 10	109300	53900	55400
1921 11	20800	9500	11300
.			
.			
.			
.			
2021 00	35381700	17531400	17850400
2021 01	1734800	890600	844300
2021 02	1783600	915200	868400
2021 03	1815700	933200	882400
2021 04	3907700	2003300	1904500
2021 05	4786400	2430300	2356100
2021 06	4843400	2449200	2394200
2021 07	4711600	2374100	2337400
2021 08	5127900	2545900	2582000
2021 09	3886100	1859600	2026500
2021 10	1938400	854400	1084100
2021 11	846100	275700	570500

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