

Paper 120-29

Creating Maps with SAS/GRAPH® - Drill Downs, Pop-Ups, and Animation

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

ABSTRACT

Maps can be created with SAS by using the GMAP procedure, one of the PROCs available within SAS/GRAPH. Recent versions of SAS/GRAPH enable users to create maps in which map areas can be hyperlinked to additional information (drill-downs); additional information can be provided to map users when a mouse cursor passes over a map area (pop-ups); and a time dimension can be added using animation. This workshop first introduces you to the basics of creating maps with SAS/GRAPH. The presentation then focuses on choropleth maps, which are two-dimensional maps that use color or fill patterns to represent data in map areas. After a quick look at basics, the majority of the workshop will be spent on learning how to move from static displays to those that offer drill-downs, pop-ups, and animation. Though the workshop will use maps to illustrate these relatively new features, concepts learned in the workshop can also be applied to graphics created with other SAS/GRAPH PROCs such as PROC GPLOT and PROC GCHART.

INTRODUCTION

The initial exercises in this workshop are designed to: introduce you to some of the basic capabilities of PROC GMAP by showing you how to present the same data in a variety of maps; give you some knowledge of the location and content of map data sets; demonstrate the difference in using map data containing projected versus unprojected X-Y coordinates; show how to select colors and gray scales for choropleth maps; show you how to produce an outline map, i.e. a map just showing map boundaries and no data. These ideas will be demonstrated using static maps. The remaining exercises demonstrate how to direct a graphics output to GIF and PDF files, and how to use a number of new features that have been added to SAS/GRAPH that enable drill-downs, pop-ups, and animation. They may be thought of as web-enabling features in that many of them are of use in the context of posting your maps on the web.

These exercises just skim the surface of what can be accomplished with SAS/GRAPH. They are meant to introduce you to features, not have you understand the features in detail. Hopefully they will pique your interest enough to learn the details.

THE DATA SETS

The following data set is used in most of the exercises. The data are from Statistics Canada. Two variables are added to the data within the data step: the percentage change in population within each province between 1996 and 2001; the population density in each province in units of population per square kilometer. The data for Nunavut are not separated from that of the Northwest Territories.

```

*** data from STATISTICS CANADA web site;
data canada_pop;
input
  province          : $2.
  province_name &  : $21.
  (pop_2001 pop_1996 area_sqk) (: comma.)
;

pct_change = 100 * (pop_2001 - pop_1996) / pop_1996;
pop_density = pop_2001 / area_sqk;

label
  pop_2001      = 'POPULATION 2001'
  pop_1996     = 'POPULATION 1996'
  area_sqk     = 'AREA SQUARE KILOMETERS'
  pct_change   = '% CHANGE IN POPULATION 1996 TO 2001'
  pop_density  = 'POPULATION PER SQUARE KILOMETER'
;

format pop_2001 pop_1996 area_sqk comma10.;

datalines;
10 NEWFOUNDLAND          512,930          551,792          370,502
11 PRINCE EDWARD ISLAND  135,294          134,557           5,684
12 NOVA SCOTIA           908,007          909,282          52,917
13 NEW BRUNSWICK         729,498          738,133          71,356

```

24	QUEBEC	7,237,479	7,138,795	1,357,743
35	ONTARIO	11,410,046	10,753,573	907,656
46	MANITOBA	1,119,583	1,113,898	551,938
47	SASKATCHEWAN	978,933	990,237	586,561
48	ALBERTA	2,974,807	2,696,826	639,987
59	BRITISH COLUMBIA	3,907,738	3,724,500	926,492
60	YUKON	28,674	30,766	474,707
61	NORTHWEST TERRITORIES	64,105	64,402	3,066,569

```

;
run;

```

In addition to the above data set, PROC GMAP requires data sets that define the boundaries of areas that are mapped. SAS/GRAPH provides map data sets that allow one to create one or more maps of nearly every country in the world. All the maps are located in a MAPS subdirectory under the SAS root directory. The libref MAPS can be used at any time to draw maps with any of the SAS-supplied map data sets without having to use a LIBNAME statement. Note, SAS/GRAPH maps must be installed to use the SAS-supplied maps.

FOUR MAPS OF CANADIAN PROVINCES (EXERCISES 1-4)

The following examples illustrate the four basic map types available with PROC GMAP. Nunavut is not shown separated from the Northwest Territories since current SAS/GRAPH maps of Canada contain no boundary definitions for Nunavut.

EX01.SAS Choropleth

The first exercise produces a choropleth map of Canadian provinces showing the distribution of population in 2001, a map where map areas are differentiated by color.

```

*** reset all graphics options to default values;
options reset=all;

*** create a choropleth map with a title and footnote;
title 'CANADIAN POPULATION BY PROVINCE, 2001';
footnote j=r 'EX01.SAS ';
proc gmap data=canada_pop map=maps.canada2;
id province;
choro pop_2001;
run;
quit;
title;
footnote;

```

Production of a map with the GMAP procedure requires both a response and a map data set. The response data set contains the data that are to be displayed in a map, while the map data set contains the information needed to draw map boundaries (in this example, Canadian provinces). PROC GMAP uses the variable in the ID statement in to relate the data to a specific geographic area. If one merges two data sets with a data step merge and a by statement, the variable(s) in the by statement must have the same type in both data sets. This is also true for the variable in the ID statement, i.e. it has the same type in both the response and map data sets. The variable in the CHORO statement determines that data to be displayed in the map.

Notice how many different colors were used to fill the map areas, i.e. four. There are a number of different options that you can use on the CHORO statement in PROC GMAP to control the number of colors that will be used. If you do not use any of these options, SAS/GRAPH will use a formula that involves the number of unique map areas to determine how many different colors to use. The decision is not based on the values in your data (the response data set). If you want to control the number of colors, you can use the LEVELS option on the CHORO statement or you can use the DISCRETE option in combination with a FORMAT statement...

```

proc gmap data=canada_pop map=maps.canada2;
id province;
choro pop_2001 / levels=3;
run;
quit;

```

Or...

```

proc format;
value popfmt
low -< 1000000 = '< ONE MILLION'
1000000 - high = 'ONE MILLION +'
;
run;

```

```
proc gmap data=canada_pop map=maps.canada2;
  id province;
  choro pop_2001 / discrete;
  format pop_2001 popfmt.;
run;
quit;
```

EX02.SAS Prism

The second exercise uses the same data, but produces a prism map, a map that uses both color and raised map areas to display population data.

```
*** reset all graphics options to default values;
goptions reset=all;

*** create a prism map with a title and footnote;
title 'CANADIAN POPULATION BY PROVINCE, 2001';
footnote j=r 'EX02.SAS ';
proc gmap data=canada_pop map=maps.canada2;
  id province;
  prism pop_2001;
run;
quit;
title;
footnote;
```

The only difference in the SAS code used to create the choropleth and prism maps is the change of the word CHORO to PRISM in the GMAP procedure. Once again, SAS chooses the number of levels to differentiate among provincial populations. The SAS code can be modified to change the viewing angle of the map.

```
title 'CANADIAN POPULATION BY PROVINCE, 2001';
footnote j=r 'EX02.SAS ';
proc gmap data=canada_pop map=maps.canada2;
  id province;
  prism pop_2001 / xview=-5 yview=-3 zview=10 ylight=2;
run;
quit;
title;
footnote;
```

The XVIEW, YVIEW, and ZVIEW options on the PRISM statement change the viewing position from the default value of a position above and to the south of the center of the map (values of .5, -2.0, and 3.0 for the X, Y, and Z views respectively). There is also an imagined light source that controls the appearance of the edges of the map areas. That is modified is using either or both the XLIGHT and YLIGHT options.

EX03.SAS Block

The third exercise uses the same data, but produces a block map, a map that uses a block in the middle of each map area to display population data.

```
*** reset all graphics options to default values;
goptions reset=all;

*** create a block map with a title and a footnote;
title 'CANADIAN POPULATION BY PROVINCE, 2001';
footnote j=r 'EX03.SAS ';
proc gmap data=canada_pop map=maps.canada2;
  id province;
  block pop_2001;
run;
quit;
title;
footnote;
```

The only difference in the SAS code used to create the choropleth and block maps is changing the word CHORO to BLOCK in the GMAP procedure. Once again, SAS chooses the number of levels to differentiate among provincial populations. The SAS code can be modified resulting in blocks whose heights are scaled continuously from the lowest to highest population values.

```
pattern1 v=s c=red r=12;
pattern2 v=ms c=green;
```

```

title 'CANADIAN POPULATION BY PROVINCE, 2001';
footnote j=r 'EX03.SAS ';
proc gmap data=canada_pop map=maps.canada2;
id province;
block pop_2001 / levels=12 shape=cylinder blocksize=5
                 xview=1.8 yview=-1.0 zview=1.0
                 nolegend coutline=black;

run;
quit;
title;
footnote;

```

PATTERN statements are used to specify that all blocks are red (12 provinces) and that the map areas are light gray. A number of options are used on the BLOCK statement. Twelve different levels are requested (one per province) and the shape and size of the blocks are changed. Just as was done with the PRISM map, the viewing angle is changed. Since the block height, not color, displays population information, the legend is suppressed (NOLEGEND). By default, map areas are outlined in their own color (in this example, green) so the COUTLINE option is used to specify a color for provincial boundaries.

EX04.SAS Surface

The final map type is a surface map. It differs from the other map types in that no map areas are displayed. Rather, a spike at the center of each map area displays map data.

```

*** reset all graphics options to default values;
options reset=all;

*** create a surface map with a title and footnote;
title 'CANADIAN POPULATION BY PROVINCE, 2001';
footnote j=r 'EX04.SAS ';
proc gmap data=canada_pop map=maps.canada2;
id province;
surface pop_2001;
run;
quit;
title;
footnote;

```

The surface map is similar to the last block map that was created (with all population 12 levels shown) in that the height of the spike shows relative values of population. The map is intended to show a gradient in population rather than exact values. There are a number of options that alter the appearance of a surface map.

```

title 'CANADIAN POPULATION BY PROVINCE, 2001';
footnote j=r 'EX04.SAS ';
proc gmap data=canada_pop map=maps.canada2;
id province;
surface pop_2001 / rotate=80 tilt=45 nlines=100 constant=30;
run;
quit;
title;
footnote;

```

The ROTATE and TILT options rotate the map about the z-axis and x-axis respectively (the default is 70 degrees for both options). The NLINES option controls the number of lines used on the map surface (default 50), while the CONSTANT option controls the appearance of the spikes (both width and height).

MAP DATA SETS (EXERCISES 5-8)

As stated earlier, SAS/GRAPH provides map data sets that allow one to create one or more maps of nearly every country in the world. At a minimum, a map data set contains three variables: X and Y (both numeric) that contain the coordinates of map areas (commonly, but not necessarily longitude and latitude); a variable that contains a value for the geographic area associated with each pair of X-Y variables. Two other variables that may be present in a map data set are SEGMENT and DENSITY. SEGMENT is used by the GMAP procedure to draw map areas that may comprise more than one polygon. DENSITY can be used to reduce the number of X-Y coordinates used to draw map boundaries when less detailed maps are permissible.

EX05.SAS The METAMAPS Data Set

In addition to map data sets, the MAPS library contains a data set (METAMAPS) with information about maps in the library.

```
*** display a list of maps in the MAPS library;
proc print data=maps.metamaps label;
var country memname;
where memtype eq 'DATA' and type eq 'GRAPH';
run;
```

The MAPS library contains both data sets and catalogs. If SAS/GIS is also installed, some of the data sets in the MAPS library are of use only with that product. The WHERE statement limits the listing to map data sets of use with SAS/GRAPH procedures.

EX06.SAS Map Data Sets and Map Information Data Sets

The output of example EX05.SAS shows that most countries are represented by two data sets, one of which ends with the number 2. The data set that ends with a 2 is not a map data set, but a data set with information about geographic areas in the companion map data set. For example, there are FRANCE and FRANCE2 data sets.

```
*** information about map areas in the FRANCE map data set;
proc print data=maps.france2;
run;
```

Output from example EX06.SAS shows the values of the variable ID that is also in the FRANCE map data set. It also shows the names of the regions in France associated with each value of ID. The variable ID in the FRANCE map data set performs the same function as the variable PROVINCE in examples 1 through 4, i.e. it is used to match data in a response data set to areas in the map data set.

EX07.SAS Map Data Sets for Canada and the United States

Two exceptions to the presence of two data sets per country in the MAPS library are Canada and the United States.

```
*** display a list of maps in the MAPS library for Canada and the United States;
proc print data=maps.metamaps label;
var country memname;
where memtype eq 'DATA' and type eq 'GRAPH' and country in ('CANADA' 'UNITED STATES');
run;
```

There are four map data sets for Canada: CANADA, CANADA2, CANADA3, CANADA4. The data sets enable mapping at both the province and region within province level. The data set CANCENS, not CANADA2, contains information about map areas: province numbers and names, region numbers and names. This is the only exception to the rule of a data set in the maps library ending with a 2 being a map information data set. CANADA2 is a map data set.

There are five map data sets for the United States: COUNTIES, COUNTY, STATES, US, USCOUNTY. The data sets enable mapping at both the state or county within state level. The data sets US2 and CNTYNAME contain information about map areas: state numbers and names, county numbers and names.

EX08.SAS Create a Map Data Set from a Shapefile

In addition to the SAS-supplied map data sets in the MAPS library, version 9 has a new procedure, PROC MAPIMPORT, that allows one to convert shapefiles to map data sets. Shapefiles are a common method of storing geographic information (for example area boundaries, streets, landmark locations) for use in a geographic information system. There are a number of shapefiles posted on the web for free download.

```
*** create a map data set from a shapefile;
proc mapimport
datafile='z:\tr38_d00.shp'
out=nd_tracts;
select county tract segment x y;
run;
```

The DATAFILE raw data file (TR38_D00.SHP) is a census tract shapefile of North Dakota that was obtained from the US Census Bureau web site. The OUT data set (ND_TRACTS) is a map data set of North Dakota census tracts. The SELECT statement restricts the variables in the map data set to only those needed to draw the map.

PROJECTED AND UNPROJECTED MAP DATA SETS (EXERCISES 9-11)

There are four map data sets of Canada. Two can be used to draw maps of Canadian provinces (CANADA2 and CANADA4), while the other two can be used to draw maps of census districts within Canadian provinces (CANADA and CANADA3). Only provincial maps have been produced thus far. What is the difference between the CANADA2 and CANADA4 data sets?

EX09.SAS Contents of CANADA2 and CANADA4 Map Data Sets

PROC CONTENTS can be used to examine the differences between the two Canadian provincial map data sets.

```
proc contents data=maps.canada;
run;

proc contents data=maps.canada4;
run;
```

CANADA2 - 4,302 observations				
#	Variable	Type	Len	Label
1	PROVINCE	Char	2	Province Abbreviation
2	SEGMENT	Num	5	Province Segment Number
3	X	Num	6	X Coordinate
4	Y	Num	6	Y Coordinate

CANADA4 - 31,311 observations				
#	Variable	Type	Len	Label
2	DENSITY	Num	5	Density for Lower Resolution Maps
1	PROVINCE	Char	2	Province Abbreviation
3	SEGMENT	Num	5	Province Segment Number
4	X	Num	6	Unprojected Longitude in Radians
5	Y	Num	6	Unprojected Latitude in Radians

There are two differences: CANADA4 contains over seven times the number of X-Y coordinates as CANADA2; CANADA4 coordinates are unprojected. The easiest way to see the consequence of using unprojected map coordinates is to create a map using the CANADA4 map data set.

EX10.SAS A Map with Unprojected CANADA4 Map Coordinates

This example uses the same response data set as example EX01.SAS, but uses map data set CANADA4.

```
*** reset all graphics options to default values;
goptions reset=all;

*** create a choropleth map with a title and footnote;
title 'CANADIAN POPULATION BY PROVINCE, 2001';
footnote j=r 'EX10.SAS ';
proc gmap data=canada_pop map=maps.canada4;
id province;
choro pop_2001;
run;
quit;
title;
footnote;
```

There are several notable differences between the output of EX01.SAS and EX10.SAS: the map is backwards; the shape is distorted; there is much more detail in the map boundaries. Another SAS/GRAPH procedure must be used prior to PROC GMAP to project the unprojected map coordinates.

EX11.SAS A Map with Projected CANADA4 Map Coordinates

PROC GPROJECT is one of a number of SAS/GRAPH procedures that are used in conjunction with PROC GMAP.

```
*** project the CANADA4 map data set;
proc gproject data=maps.canada4 out=c4_projected;
id province;
run;

*** create a choropleth map with a title and footnote;
title 'CANADIAN POPULATION BY PROVINCE, 2001';
footnote j=r 'EX11.SAS ';
```

```
proc gmap data=canada_pop map=c4_projected;
id province;
choro pop_2001;
run;
quit;
title;
footnote;
```

COLORS AND GRAY SCALES IN CHOROPLETH MAPS (EXERCISES 12-13)

Previous choropleth maps have used SAS-selected colors. Map colors can also be user-selected and colors may be specified in a number of different ways.

EX12.SAS Create a Choropleth Map with User-Defined Colors

PATTERN statements can be used to override the default color selection made by PROC GMAP. Also, a LEGEND statement can be used to move the map legend, leaving more room to draw the map.

```
*** reset all graphics options to default values;
goptions reset=all;

*** choose a set of colors - sequentially increasing shades of blue;
pattern1 v=ms c=cxeff3ff;
pattern2 v=ms c=cxbdd7e7;
pattern3 v=ms c=cx6baed6;
pattern4 v=ms c=cx2171b5;

*** specify parameters for the legend;
legend1
origin=(75,60)pct
across=1
mode=share
label=(position=top j=c 'POPULATION PER SQ KM')
shape=bar(3,4)pct
cborder=blue
;

*** create a choropleth map with user-defined colors and legend;
title 'CANADIAN POPULATION DENSITY BY PROVINCE, 2001';
footnote j=r 'EX12.SAS ';
proc gmap data=canada_pop map=maps.canada2;
id province;
choro pop_density / levels=4 outline=black legend=legend1;
run;
quit;
title;
footnote;
```

The PATTERN statements use the RGB color-naming scheme to define a series of colors that increase from light to dark blue. The characters CX in the PATTERN statement specify that the RGB color scheme is being used. The hexadecimal numbers following CX indicate the degree of red, green, and blue that comprise the color and can range from 00 to FF (0 to 255). The RGB method is just one of several ways to select colors in SAS. Other methods include: CMYK, HLS, HSV (or HSB), and gray-scale color codes; SAS and CNS color names. There is a web site that makes it easy to select color schemes for choropleth maps and it was used to select the colors used in this example...

<http://www.colorbrewer.com>

The LEGEND statement specifies that the origin (lower left-hand corner of the box around the legend) is in the upper-right portion of the output area (75 and 60 percent in the x and y directions respectively). More options change other characteristics of the legend. The LEGEND= option on the CHORO statement in PROC GMAP adds the modified legend to the map.

EX13.SAS Create a Choropleth Map with User-Defined Gray Scales

In some situations, for example where maps must be copied, it may be more appropriate to create a map not in color but in blacks, whites, and grays. The same data as used in example EX09.SAS are used in the following SAS code.

```
*** choose a set of colors - sequentially increasing shades of gray;
pattern1 v=ms c=grayf4;
pattern2 v=ms c=grayd4;
```

```

pattern3 v=ms c=graya4;
pattern4 v=ms c=gray64;

*** specify parameters for the legend;
legend1
origin=(75,60)pct
across=1
mode=share
label=(position=top j=c 'POPULATION PER SQ KM')
shape=bar(3,4)pct
cborder=black
;

*** create a choropleth map with user-defined colors and legend;
title 'CANADIAN POPULATION DENSITY BY PROVINCE, 2001';
footnote j=r 'EX13.SAS ';
proc gmap data=canada_pop map=maps.canada2;
id province;
choro pop_density / levels=4 coutline=black legend=legend1;
run;
quit;
title;
footnote;

```

Gray-scale colors are defined using hexadecimal numbers. Grays can range from GRAYFF (white) to GRAY00 (black). The pattern definitions used in this example range from light to dark gray.

OUTLINE MAPS (EXERCISES 14-15)

All maps created this far have displayed data contained in a response data set. There are occasions when a map showing only map boundaries (no data) is needed.

EX14.SAS Create an Outline Map of Nova Scotia with Census Districts

The CANADA data set is used to create an outline map of the province of Nova Scotia showing the census district boundaries within the province.

```

*** reset all graphics options to default values;
goptions reset=all;

pattern v=me;

title 'NOVA SCOTIA REGIONS';
footnote j=r 'EX14.SAS ';
proc gmap data=maps.canada (obs=1) map=maps.canada all;
where province eq '12';
id cdcode;
choro cdcode / nolegend;
run;
quit;
title;
footnote;

```

A WHERE statement is used to restrict the map to observations for the province of Nova Scotia (see example EX01.SAS for province codes). There is an additional variable within the CANADA data set that assigns each observation to a census district within a province, CDCODE. The census districts in Nova Scotia comprise 18 counties. Notice that the response and map data sets are the same. Only one observation need be used for the response data set. The default behavior of PROC GMAP is to display only map areas present in the response data set. To display all map areas, the ALL option is used.

EX15.SAS Create a More Detailed Outline Map of Nova Scotia with Census Districts

The map produced with the CANADA map data set does not have much detail in the area boundaries. The CANADA3 map data set is similar to the CANADA map data set in that it includes the variables PROVINCE and CDCODE. However, it contains many more x-y coordinates (unprojected).

```

*** reset all graphics options to default values;
goptions reset=all;

```

```

*** create a map data set with projected x-y coordinates for Nova Scotia census districts;
proc gproject data=maps.canada3 out=nova_scotia;
where province eq '12';
id cdcodes;
run;

pattern v=me;

title 'NOVA SCOTIA REGIONS';
footnote j=r 'EX15.SAS ';
proc gmap data=nova_scotia (obs=1) map=nova_scotia all;
id cdcodes;
choro cdcodes / nolegend;
run;
quit;

```

PROC GPROJECT is used to create a map data set with projected x-y coordinates. A WHERE statement limits the data set is limited to only those observations from Nova Scotia. The projected map data set is used to create the map. Notice that the map has more detail and its position has changed. The coordinates in the CANADA data set are Cartesian coordinates that are the result of projecting a map of all Canadian provinces. The view of Nova Scotia produced in example EX14.SAS is what one would 'see' from the center of the map of all of Canada. The map produced with example EX15.SAS is the view of Nova Scotia 'seen' from the center of Nova Scotia. Even if that is not completely clear, the contrast in the two maps makes it clear that maps of small portions of Canada should be produced with observations extracted from either the CANADA2 or CANADA4 data set (more x-y coordinates for detail, unprojected coordinates that allow positional accuracy).

ALTERNATIVE DESTINATIONS FOR MAP OUTPUT (EXERCISES 16-17)

All the maps produced thus far have been directed to the graphics output window in the display manager. There are a variety of different formats for maps in addition to the graphics output window. Among them GIF and PDF. Either of these formats are commonly used to post graphical output on the web.

EX16.SAS CREATE A MAP AS A GIF FILE

This example uses the data from example EX12.SAS. A map of population density in Canadian provinces is created as a GIF file. The file can easily be posted on the web or incorporated into other programs (e.g. Powerpoint).

```

*** reset all graphics options to default values, select the GIF device driver, set some options;
goptions reset=all
device=gif
gsfname=mapout
ftext='Arial/bo'
xpixels=800
ypixels=600
;

*** choose a set of colors - sequentially increasing shades of blue;
pattern1 v=ms c=cxeff3ff;
pattern2 v=ms c=cxbdd7e7;
pattern3 v=ms c=cx6baed6;
pattern4 v=ms c=cx2171b5;

*** specify parameters for the legend;
legend1
origin=(75,60)pct
across=1
mode=share
label=(position=top j=c 'POPULATION PER SQ KM')
shape=bar(3,4)pct
cborder=blue
;

filename mapout 'z:\canada.gif';

*** create a choropleth map with user-defined colors and legend;
title 'CANADIAN POPULATION DENSITY BY PROVINCE, 2001';
footnote j=r 'EX16.SAS ';

```

```

proc gmap data=canada_pop map=maps.canada2;
id province;
choro pop_density / levels=4 outline=black legend=legend1;
run;
quit;
title;
footnote;

```

After resetting all graphics options to default values, a DEVICE= option is used in the GOPTIONS statement to select the GIF device driver. Until this option is reset, all graphics output is directed to a GIF file. The GSFNAME= option species a FILEREF for the GIF file. The remaining options choose the True-Type font Arial Bold for all text (FTEXT=) and a size for the map (XPIXELS= and PIXELS=). The remaining statements are identical to those in example EX12.SAS except for the FILENAME statement that connects the FILEREF MAPOUT to a real filename.

EX17.SAS CREATE A MAP AS A PDF FILE

If the map in the GIF file is posted on the web, its appearance will vary depending on the resolution of the monitor on which it is displayed. On a high-resolution monitor, it might look quite small. An alternative format for web posting that is not sensitive to monitor resolution is PDF.

```

*** reset all graphics options to default values, set some options;
goptions reset=all
ftext='Helvetica/bo'
rotate=landscape
;

*** choose a set of colors - sequentially increasing shades of blue;
pattern1 v=ms c=cxeff3ff;
pattern2 v=ms c=cxbdd7e7;
pattern3 v=ms c=cx6baed6;
pattern4 v=ms c=cx2171b5;

*** specify parameters for the legend;
legend1
origin=(70,60)pct
across=1
mode=share
label=(position=top j=c 'POPULATION PER SQ KM')
shape=bar(3,4)pct
cborder=blue
;

options orientation=landscape;
ods listing close;
ods pdf file='z:\canada.pdf' notoc;

*** create a choropleth map with user-defined colors and legend;
title 'CANADIAN POPULATION DENSITY BY PROVINCE, 2001';
footnote j=r 'EX17.SAS ';
proc gmap data=canada_pop map=maps.canada2;
id province;
choro pop_density / levels=4 outline=black legend=legend1;
run;
quit;
title;
footnote;

ods pdf close;
ods listing;
options orientation=portrait;

```

An option in a GOPTIONS statement is used to select a font for the text, Helvetica Bold. The output delivery system (ODS) is used to create the map as a PDF file. The default orientation is portrait. An OPTIONS statement is used to change the orientation to landscape prior to the ODS statements. The statement ODS LISTING CLOSE suppresses output to the graphics output window. The next statement directs the output to a file (the NOTOC option suppresses the default insertion of a bookmark in the PDF output). Two more ODS statements after PROC GMAP close the PDF file and then reopen the output window.

NEW FEATURES (EXERCISES 18-22)

All maps produced thus far have been static, i.e. their only purpose is display and/or printing. New features within SAS/GRAPH allow for the creation of maps with dynamic features: drill-down areas; pop-ups; animation.

EX18.SAS A Drill-Down Map

There are a number of different methods for producing a map with drill-down areas. This example uses the GIF device driver plus ODS and produces a map in which each province is linked to a file. Clicking the mouse on any province will drill-down to a file containing more information about that province.

```

*** add another variable to the canada_pop data set, defining an ALT tag and a drill-down destination;
data canada_pop;
set canada_pop;
prov_link = 'ALT="' || trim(province_name) || '" href=z:\province'
           || province || '.htm';
run;

*** reset graphics options to default values, select the GIF device driver, map size, font;
goptions reset=all
device=gif
xpixels=800
ypixels=600
ftext='Tahoma/bo'
;

*** choose a set of colors - sequentially increasing shades of blue;
pattern1 v=ms c=cxeff3ff;
pattern2 v=ms c=cxbdd7e7;
pattern3 v=ms c=cx6baed6;
pattern4 v=ms c=cx2171b5;

ods listing close;
ods html path = 'z:\' (url=none) file='canada.html';

title 'CANADIAN POPULATION DENSITY BY PROVINCE, 2001';
footnote j=r 'EX18.SAS';
proc gmap data=canada_pop map=maps.canada2;
id province;
choro pop_density / levels=4 coutline=blue html=prov_link;
format pop_density 5.2;
run;
quit;
title;

ods html close;
ods listing;

```

When PROC GMAP is used, there is a new option used on the CHORO statement, HTML=. The variable PROV_LINK was added to the data set CANADA_POP in the second data step. The value of PROV_LINK is a typical bit of HTML code for a hyperlink. First, an ALT tag is defined. The value of the ALT tag is the text that will appear when the mouse cursor is placed on a map area, in this case it is the province name. Then a hyperlink is added using an HREF. The value of HREF is the name of file. A click of the mouse while on a given map area will drill down to the file named after HREF=. There are a number of other methods to create drill-downs in graphics produced with SAS/GRAPH. However, none are as simple as the combination of GIF and ODS.

EX19.SAS An ACTIVEX Map with Continuous Colors, Pop-Ups, On-Screen Editing

The ACTIVEX device driver enables a number of features. They include the ability to produce a continuously colored map (one with colors ranging from low to high intensity) with color increasing in intensity as the value of the mapped variable increase. This can be done without using any PATTERN statements. The map also has pop-up information associated with each map area.

```

*** use the canada_pop data set to create a format used in PROC GMAP to display province names;
*** first create a CNTLIN data set view, then use the CNTLIN data set in PROC FORMAT;
data prov_fmt (keep=fmtname start label) / view=prov_fmt;
retain fmtname '$provname';
set canada_pop (rename=(province=start province_name=label));
run;

proc format cntlin=prov_fmt;
run;

```

```

*** reset graphics options to default values, select the JAVA device driver and image size;
goptions reset=all
device=activex
xpixels=800
ypixels=600
;

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

title 'CANADIAN POPULATION DENSITY BY PROVINCE, 2001';
footnote j=r 'EX19.SAS';
proc gmap data=canada_pop map=maps.canada2;
id province;
choro pop_density / levels=all nolegend;
format province $provname.;
label province = 'PROVINCE';
run;
quit;
title;

ods html close;
ods listing;

```

This example produces the same map as examples EX12.SAS and EX13.SAS, provincial population density in 2001. However, there are a number of differences in the final product. The ACTIVEEX device driver is specified in a GOPTIONS statement. When used in combination with LEVELS=ALL in PROC GMAP, a map with a continuous color scheme (a unique color for each value of the mapped variable) is produced. No PATTERN statements are necessary. When viewing the map, information about each area (the value of the ID variable and the value of the variable in the CHORO statement) appear in pop-up box as the mouse cursor is moved over the map. Also, if the right mouse button is clicked anywhere on the screen display of the map, an editing menu will appear that allows a user to alter the appearance of the map.

EX20.SAS A JAVA Map with Pop-Ups, On-Screen Editing

The JAVA device driver is similar to the ACTIVEEX device driver in that pop-up information is associated with each map area and on-screen editing of map appearance is enabled. This example uses the same SAS code as example EX19.SAS up to the point of the GOPTIONS statement.

```

*** reset graphics options to default values, select the JAVA device driver and image size;
goptions reset=all
device=java
xpixels=800
ypixels=600
;

*** choose a set of colors - sequentially increasing shades of blue;
pattern1 v=ms c=cxeff3ff;
pattern2 v=ms c=cxbdd7e7;
pattern3 v=ms c=cx6baed6;
pattern4 v=ms c=cx2171b5;

ods listing close;
ods html file = 'z:\canada2.html';

title 'CANADIAN POPULATION DENSITY BY PROVINCE, 2001';
footnote 'EX20.SAS';
proc gmap data=canada_pop map=maps.canada2;
id province;
choro pop_density / levels=4 nolegend;
format province $provname. pop_density 6.3;
label province = 'PROVINCE';
run;
quit;
title;

ods html close;
ods listing;

```

The LEVELS=ALL option used in example EX19.SAS is replaced with LEVELS=4. PATTERNS are also defined for the our areas. The LEVELS=ALL option used previously does not produce a continuous color pattern with the JAVA device driver. It does produce 12 (the number of provinces) unique colors if no PATTERN statements are used to set a user-defined color

scheme. As with the ACTIVEX device driver: pop-ups appear on the screen as the mouse is moved over the map; clicking the right mouse button anywhere on the screen display results in the appearance of a menu of items that can be used to alter the appearance of the map (though a different menu structure than with ACTIVEX).

EX21.SAS A JAVAMETA Map with User-Controlled Pop-Ups

The pop-ups in examples EX19.SAS and EX20.SAS contain default information. The pop-up box contains the value of the ID variable and the value of the of the variable in the CHORO statement (both can be either formatted or unformatted). The JAVAMETA device driver enables the creation of user-defined information that will appear in the pop-up box. The output requires access to the MetaView Applet to display properly.

```

*** add another variable to the canada_pop data set, a TIP that will appear in a pop-up box;
data canada_pop;
length popvar $500;
set canada_pop;
popvar = 'tip=[' ||
    quote(province_name) || ' ' ||
    quote('POPULATION') || ' ' ||
    quote('1996: ' || put(pop_1996,comma10.)) || ' ' ||
    quote('2001: ' || put(pop_2001,comma10.)) || ' ' ||
    quote('%CHANGE: ' || put(pct_change,10.1)) || ' ' ||
    quote('POPULATION DENSITY') || ' ' ||
    quote('PER SQ KM: ' || put(pop_density,10.3)) || ']' ;

run;

*** reset graphics options to default values, select the JAVA device driver and map size;
goptions reset=all
device=javameta
xpixels=800
ypixels=600
ftext='HelveticaBold'
;

*** choose a set of colors - sequentially increasing shades of blue;
pattern1 v=ms c=cxeff3ff;
pattern2 v=ms c=cxbdd7e7;
pattern3 v=ms c=cx6baed6;
pattern4 v=ms c=cx2171b5;

*** specify parameters for the legend;
legend1
origin=(70,60)pct
across=1
mode=share
label=(position=top j=c 'POPULATION PER SQ KM')
shape=bar(3,4)pct
;

ods listing close;
ods html path='z:\'
    body='canada.html'
    parameters=("ZoomControlEnabled"="false"
               "DataTipStyle"="Stick_Fixed");

title 'CANADIAN POPULATION DENSITY BY PROVINCE, 2001';
footnote j=r 'EX21.SAS';
proc gmap data=canada_pop map=maps.canada2;
id province;
choro pop_density / levels=4 coutline=blue legend=legend1 html=popvar;
format province $provname. pop_density 6.3;
label province = 'PROVINCE';
run;
quit;
title;

ods html close;
ods listing;

```

In PROC GMAP, the HTML= option is used as was done in example EX18.SAS when a drill-down link was added to each map area. However, in this example, the variable POPVAR used with the HTML= option does not define a link. Rather, it species the content of a pop-up associated with each map area. The variable POPVAR is added in a data step and its content is a

mixture of text and the values of several variables in the data set. With previous pop-ups, there was little control over pop-up content. ODS is used in combination with the JAVAMETA device driver and the MetaView Applet. Use of the JAVAMETA device driver adds a reference to the location of the MetaView Applet when it creates the file containing commands that display graphics. There are a number of parameters one can use with this Applet. By default, a zoom control is added to graphics output when the JAVAMETA device driver is used. The ZoomControlEnabled parameter is used to disable addition of the zoom control. The method by which a pop-up box is associated with a map area is controlled by the DataTipStyle parameter, and it is set to STICK_FIXED. The zoom and tip style are only two of many available parameters.

EX22.SAS An Animated Map

Animation is a way to add a time-dimension to mapped data. A series of maps can be shown in succession showing changes in mapped data over time. This example shows the movement of the mean center of population of the United States from 1790 through 2000. The mean population center is a point at which a flat map of the United States would balance given that all people represented in the population had the same weight. The data are from the US Census Bureau and contain the latitude and longitude of the population center, plus the county and state of the center. These data are used to create an annotated data set. That data set is used to add symbols (stars) to the map as the location of the center changes over the years that are mapped.

```

*** data from US CENSUS BUREAU web site;
*** create an annotate data set to place a star on the map at the population center;
*** year, latitude/longitude of US population center, county/state of US population center;
data popctr;
retain
function 'label'
xsys
ysys      '2'
hsys      '3'
position  '5'
size      4
color     'green'
style     'marker'
text      'V'
when      'a'
;
infile datalines dsd;
input year : $4. y x co : $40. st : $2.;
fips = stfips(st);
y = y*constant('pi')/180;
x = x*constant('pi')/180;
datalines;
1790,39.275,76.18667      ,KENT,MD
1800,39.26833,76.94167    ,HOWARD,MD
1810,39.19167,77.62      ,LOUDON,VA
1820,39.095,78.55       ,HARDY,WV
1830,38.965,79.28167    ,GRANT,WV
1840,39.03333,80.3       ,UPSHUR,WV
1850,38.98333,81.31667   ,WIRT,WV
1860,39.00667,82.81333   ,PIKE,OH
1870,39.2,83.595        ,HIGHLAND,OH
1880,39.06889,84.66111   ,BOONE,KY
1890,39.19889,85.54806   ,DECATUR,IN
1900,39.16,85.815        ,BARTHOLOMEW,IN
1910,39.17,86.53889      ,MONROE,IN
1920,39.1725,86.72083    ,OWEN,IN
1930,39.0625,87.135      ,GREENE,IN
1940,38.948333,87.37639  ,SULLIVAN,IN
1950,38.80417,88.36889    ,CLAY,IL
1960,38.59944,89.20972   ,CLINTON,IL
1970,38.46306,89.70611   ,ST CLAIR,IL
1980,38.13694,90.57389   ,JEFFERSON,MO
1990,37.87222,91.21528   ,CRAWFORD,MO
2000,37.69699,91.80957   ,PHELPS,MO
;
run;

*** combine observations for selected states from MAPS.STATES map data set with the annotate data;
data both;
set maps.states (where=(fipstate(state) in ("IA" "IL" "IN" "OH" "PA" "MO" "KY" "WV" "VA" "TN" "NC"
"MD" "DE" "DC" "AR" "NJ" "NY")) and density le 3)
popctr;
run;

```

```

*** project the unprojected latitude/longitude in data set BOTH;
proc gproject data=both out=bothproj;
id state;
run;

*** separate projected map coordinates from the projected annotate data coordinates;
data pmap pop;
set bothproj;
if when eq 'a' then output pop;
else          output pmap;
run;

*** reset all graphics options to default values, select the GIFANIM device driver;
*** specify other parameters for the GIF file including ITERATION and DELAY (animation controls);
goptions
reset=all
device=gifanim
gsfname=animout
gsfmode=replace
gunit=pct
ftext='Tahoma'
ctext=blue
border
iteration=0
delay=200
;

*** ;
pattern v=e c=blue;

%macro manymaps;
%do i=1 %to 22;

%if &i eq 2 %then goptions gsfmode=append;;
%if &i eq 22 %then goptions gepilog='3B'x;;

data anno;
length text $50;
set pop (obs=&i) end=last;
if last then do;
    color='red';
    call symput('year',year);
    call symput('cost',trim(co) || ' COUNTY, ' || st);
    call symput('fips',put(fips,z3.));
end;
run;

pattern v=s c=yellow;

title1 h=3;
title2 h=5 "CENTER OF US POPULATION:  &YEAR";
footnote1 h=3 "&cost";
footnote2 h=10;
proc gmap map=pmap data=pmap (where=(state=&fips)) all;
id state;
choro state / discrete nolegend coutline=blue cempty=blue annotate=anno;
run;
quit;
title;
footnote;
%end;
%mend;

filename animout 'z:\popctr.gif';
%manymaps;

```

This is a complicated example in that it uses an annotate data set and macro code. The annotate data set is the only way to add symbols to a map. Since twenty-two maps are being created, use of a macro cuts down on the amount of SAS code that is written. The purpose of this example is not to understand annotation and/or macros. Rather, it is to show that the GIFANIM device driver can produce an animated GIF file. When the GIF file is displayed, the twenty-two maps are shown in succession. The DELAY= option in the GOPTIONS statement controls the time between changing from one map to another and it is expressed in .01 seconds. The DELAY=200 in this example shows each map for 2 seconds. The ITERATION= option set to 0

means that the display will repeat the display of maps until stopped by the user. The JAVAMETA device driver and MetaView Applet parameters can also produce an animated display and it can have on screen controls for paging through the maps and for controlling the time delay between maps.

RESOURCES

Just as with other SAS/GRAPH procedures, PROC GMAP can produce 'publication quality' results. If you want to display data on a state or county basis, the needed maps are supplied with SAS/GRAPH, as are those needed to produce maps of many different countries. Information on SAS-supplied maps can be found at...

<http://support.sas.com/rnd/datavisualization/mapsonline/html>

Examples of PROC GMAP output and the sample code can be found at...

<http://ftp.sas.com/techsup/download/sample/graph/gmap-examples-list.html>

For more information about the METAVIEW applet and the SAS code used in several example 21, look at...

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

If you want to create maps of other areas, e.g. census tracts or zip-based maps, you will have to create your own map data sets or purchase them. Some vendors sell files already in SAS map data set format. If you want more information on vendors, go to the SAS Institute web site...

<http://support.sas.com/faq/004/FAQ00400.html>

If you are using Version 9 of SAS, a new procedure, PROC MAPIMPORT, allows you to convert files in ESRI shapefile format into SAS/GRAPH map data sets. Information on this procedure is available in V9 help, and also at...

<http://support.sas.com/rnd/datavisualization/papers/SASMapping.pdf>

Many ESRI shapefiles are posted for free download at the US Census Bureau web site...

http://www.census.gov/geo/www/cob/bdy_files.html

As mentioned earlier the web site...

<http://www.colorbrewer.com>

is designed to assist you in selecting colors for maps.

Finally, there is a book, *Maps Made Easy Using SAS...*

<http://www.sas.com/apps/pubscat/bookdetails.jsp?catid=1&pc=57495>

that expands on the material covered in this paper, including a chapter on creating maps for posting on the web.

CONCLUSION

PROC GMAP either by itself or in combination with other SAS/GRAPH procedures allows you to create a number of different types of maps. The maps can be static, or can have new features such as drill-down links, pop-up boxes with additional information about map areas, and animation. Though the new features are demonstrated with PROC GMAP, the techniques are applicable to other graphics procedures.

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

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

APPENDIX

Selected maps produced by examples in the workshop are displayed here (the example associated with each map can be found in the footnote).







