

Paper 194-2009

Create State and County Plots Easily Using a Set of SAS/Graph® Utility Macros

James L. Ribble, Columbus Technologies, Inc., Atlanta, Ga.

ABSTRACT

This paper will introduce a set of utility macros designed to simplify the creation of state and county level data plots. By making use of macros with execution-time parameters these routines can be used by even novice SAS users to create geographical data plots with little or no knowledge of SAS/Graph. The macros demonstrated will also allow users to overlay text and data-dependant markers on their plots as well as combine multiple plots into a single graphic image.

INTRODUCTION

Creating state and county data plots from multiple data sources can often be a redundant process. It can also seem less than clear to other SAS programmers or casual SAS users unfamiliar with the SAS/Graph product. In this paper I will outline a basic set of SAS macros which can be used to create these plots with a minimum of coding or knowledge of SAS/Graph. These macros will allow the user to create U.S., state and county level plots as well as add text, color-coded location markers and imported graphic images. A macro is also included which will allow multiple plots to be combined or superimposed on one another within a single document.

USAGE CONCEPTS

First one or more macros are called using a set of parameters designed to customize the routine's output. The individual outputs are stored in a graphic catalog under user supplied names. After all the desired graphic items are created they can each be referenced by name using the %OUTPUTPLOT macro which sends the specified graphic to a GIF output file. Alternatively the %COMBINEPLOTS macro could be used to composite multiple graphics into a single image. This image could subsequently be combined with additional graphics or sent to an output file via the %OUTPUTPLOT macro.

THE MACROS

Generally speaking the logic in the macros simply performs text substitution within blocks of SAS/Graph code thus saving the user from having to write the SAS/Graph logic themselves. The macro parameters provide the text to be used in these substitutions. What follows is a brief overview of each macro, its function and basic parameters.

%SETUP

This macro sets up the appropriate environment and is executed prior to calling any of the other macros. It has no parameters.

%CLEANUP

This macro performs some basic house cleaning and is called after you are finished creating your plots. It deletes the temporary graphics catalogs and files used by the various plot macros. Like the %Setup macro it called without any parameters.

%CREATEMAP

This macro is the real workhorse. With this one macro both state and county level plots can be created. Calling the macro with all of its parameters it looks like this:

```
%CreateMap(name,
           states=,
           data=,
           var=,
           by=,
           statevar=,
           countyvar=,
           transparency=,
           backgroundcolor=,
           format=,
           colorscheme=,
           legend=,
           datalevels=,
           markers=)
```

These parameters control the following:

<u>PARAMETER</u>	<u>EXPLANATION</u>	<u>DEFAULT</u>
NAME	Defines the name the created graphic should be referenced by.	
STATES	A list of state FIPS codes for the states to be plotted (separated by spaces).	All continental US
DATA	The name of the data member containing the values to be plotted.	
VAR	The data variable to be plotted.	
BY	Specifies whether the plot is by state or county.	
STATEVAR	The variable containing the state FIPS code.	STATE
COUNTYVAR	The variable containing the county FIPS code.	COUNTY
TRANSPARENCY	Y (yes) or N (no) if background transparency is desired.	Y
BACKGROUNDCOLOR	The background color.	WHITE
FORMAT	The format to be applied to the data variable.	
COLORSCHEME	The shading color to be used for the plot.	BROWN
LEGEND	Y (yes) or N (no) if a legend is desired.	Y
DATALEVELS	The number of discrete data levels present in the data variable.	
MARKERS	Optional. The name of a "Markers" (Annotate) data member created with the %ADDMARKERS macro.	

A plot will be created for the states listed in the STATES parameter. If the BY=COUNTY option is supplied individual counties within the state will be plotted. The COLORSCHEME parameter specifies a color base for the plot, e.g. brown, blue, green, etc. This color will be used to shade the plotted data values from light to dark based on the number of data levels specified by the DATALEVELS value. Although the DATALEVELS value could have been derived programmatically requiring it as a parameter helps the user ensure that the shading levels are consistent across multiple plots which may be combined on a single page regardless of any differences in their datasets.

%ADDTEXT

The %AddText macro creates a graphic image containing a single line of text:

```
%AddText ( name ,
           text=,
           x=,
           y=,
           color=,
           height=,
           font=)
```

The %AddText parameters control:

<u>PARAMETER</u>	<u>EXPLANATION</u>	<u>DEFAULT</u>
NAME	Defines the name the created graphic should be referenced by.	
TEXT	The text to be made into a graphic image.	

X	The horizontal position of the text specified as a percentage of the graphic area.	50
Y	The vertical position of the text specified as a percentage of the graphic area.	50
COLOR	The text color.	BLACK
HEIGHT	The text height specified as a percentage of the graphic area.	5
FONT	The text font.	SIMPLEX

%ADDIMAGE

The %AddImage macro allows an external graphic image, a company logo for example, to be imported for inclusion in the output document:

```
%AddImage ( name ,
            image= )
```

The %AddImage macro has only two parameters:

<u>PARAMETER</u>	<u>EXPLANATION</u>	<u>DEFAULT</u>
NAME	Defines the name the created graphic should be referenced by.	
IMAGE	The fully qualified path and name of the graphic file to be imported.	

%ADDMARKERS

%AddMarkers allows data dependant, color-coded makers to be overlaid on a geographical plot to mark cities or other locations. The color of these markers can be made to match the legend of the map which they will overlay. The output from this routine is not a graphic image but rather a SAS Annotate dataset that can be fed into the %CreateMap macro via its optional "MARKERS" parameter. The markers will then be included in the %CreateMap graphic output. The macro looks like this:

```
%AddMarkers (Name ,
             Data= ,
             var= ,
             states= ,
             lat=lat ,
             long=long ,
             statevar= ,
             size= ,
             color= ,
             datalevels= ,
             colorscheme=)
```

Its parameters are:

<u>PARAMETER</u>	<u>EXPLANATION</u>	<u>DEFAULT</u>
NAME	Specifies the name the created markers (Annotate) dataset should be referenced by.	
DATA	The name of the data member containing the values to be plotted.	
VAR	The data variable to be plotted.	

STATES	A list of state FIPS codes for the states to be plotted (separated by spaces).	All continental US
LAT	The name of the variable containing the latitude of the marker to be plotted.	LAT
LONG	The name of the variable containing the longitude of the marker to be plotted.	LONG
STATEVAR	The variable containing the state FIPS code.	STATE
SIZE	The marker size specified as a percentage of the graphic area.	1.1
COLOR	A fixed marker color or the word 'DATA' if a color scheme is to be used (see the COLORScheme parameter).	BLACK
DATALEVELS	The number of discrete data levels present in the data variable.	
COLORScheme	The base color to be used in creating a graduated color scheme for the plot.	BROWN

%COMBINEPLOTS

%CombinePlots can be used to combine or overlay multiple graphic items created by the other macros into a single image like this:

```
%CombinePlots (Name ,
                GRAPHIC1[x1,y1,x2,y2] , ... GRAPHICx[x1,y1,x2,y2])
```

The parameter format for this macro is a bit different from the other macros. The logic uses the macro PARMBUFF options to parse out a variable number of parameter sets:

<u>PARAMETER</u>	<u>EXPLANATION</u>	<u>DEFAULT</u>
NAME	Defines the name the created graphic should be referenced by.	
GRAPHICx[X1,Y1,X2,Y2]	Define 1 to x graphics to be included, their plot size and location on the output page separated by commas. Each parameter set takes the form: NAME[lower-x,lower-y,upper-x,upper-y] The x and y locations are specified as a percentage of the graphic area.	

Any number of previously created graphics can be specified. The corresponding X and Y values specify the location, size and shape of the graphic on the composited output page. The newly created graphic is then store by the specified name.

%OUTPUTPLOT

%OutputPlot is quite simple and does exactly what its name implies, it output a finalized graphic image to a GIF output file. It too has only two parameters:

```
%OutputPlot (name ,
              dest=)
```

Its two parameters perform the following functions:

<u>PARAMETER</u>	<u>EXPLANATION</u>	<u>DEFAULT</u>
NAME	Specifies the name of the previously created graphic to be output.	
DEST	The fully qualified path and name of the GIF file to be created.	

USAGE EXAMPLE #1

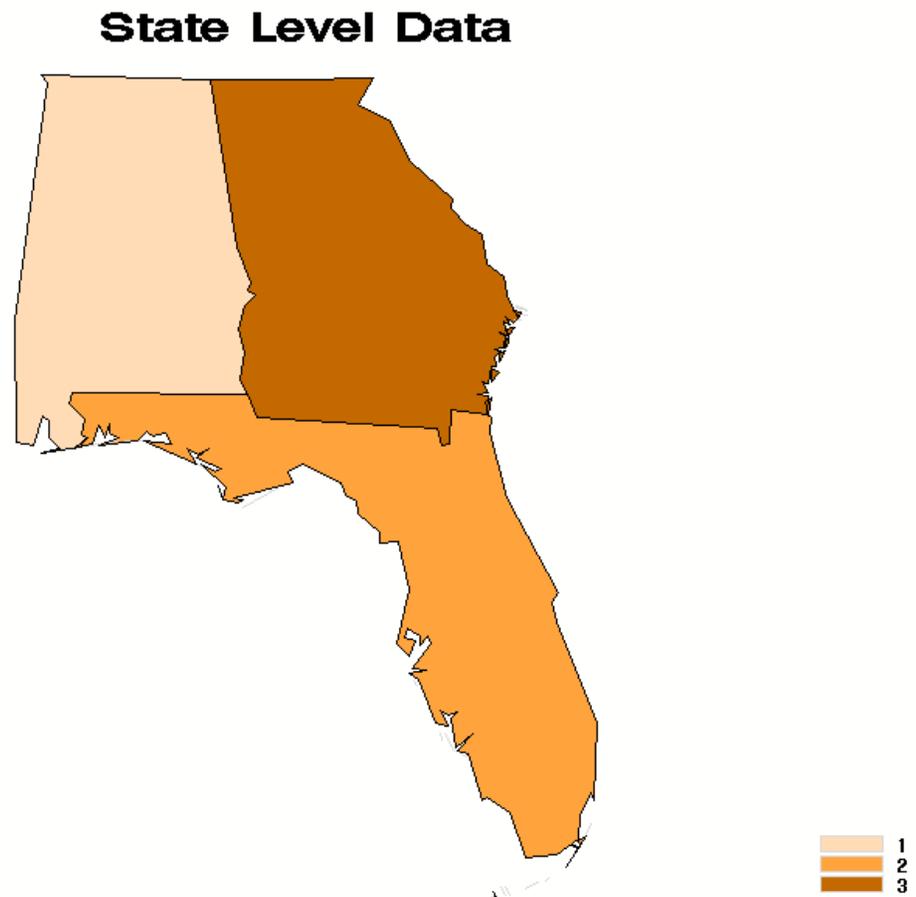
This will be a simple plot of state level data for Alabama, Georgia and Florida. Note the use of the TITLE statement and the "GOPTIONS RESET" statement to clear it after the first macro call. The reset is needed to ensure that this TITLE statement is not added to the output of any subsequent steps.

THE CODE

```
%Setup
title1 .f=swissxb .c=black .h=2 "State Level Data";
%CreateMap(AlGaFl,
           datalevels=3,
           states=1 12 13,
           data=statedata,
           var=xyzzzy,
           statevar=_state,
           by=state)
goptions reset=(title footnote);
%OutputPlot(AlGaFl,
            dest=C:\Example1.gif)
%Cleanup
```

THE RESULTS

Our output looks like this:



Note the light-to-dark shading performed automatically by the %CreateMap macro.

USAGE EXAMPLE #2

Now we will plot the same states using county level data, add the SAS logo to the page as well as a data dependent, color coded marker for Atlanta. Our color scheme this time will be green. First the %Setup macro is run to set up our environment. We then create a marker dataset for Atlanta and add it to the map output created by the %CreateMap macro. Since the %AddMarkers and %CreateMap macros both use the same color base and have the same number of data levels specified and the COLOR=DATA option was specified in the %AddMarkers macro call the color of the Atlanta marker will map to the legend created for the state plot. The reason that the states to be plotted must be specified in the %AddMarkers macro is to ensure that the code's projection of the marker's latitude and longitude will sync with the states to be plotted by the %CreateMap macro. Next the %AddImage macro imports a copy of the SAS logo. %CombinePlots is then used to combine the state plot and SAS logo into a single graphic named 'Composite'. The %OutputPlot macro then creates our GIF output file from the 'Composite' image. Finally the %Cleanup macro deletes the temporary files created by the other macros.

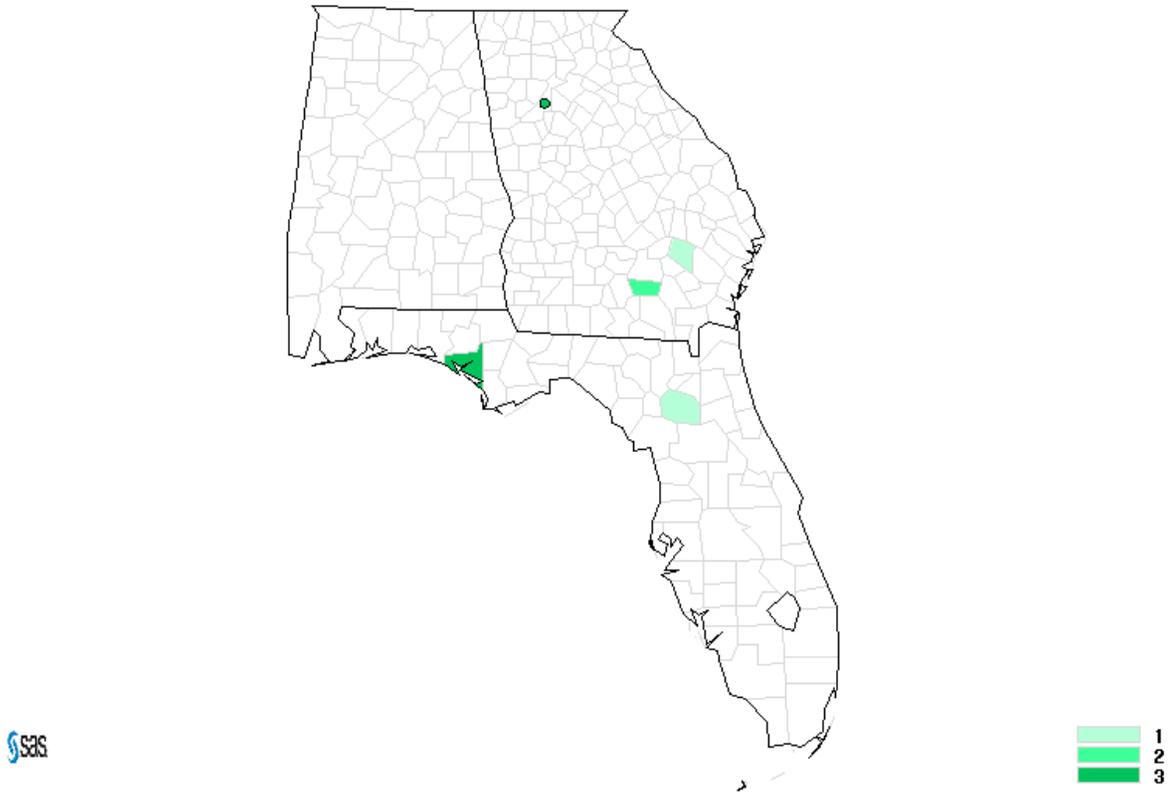
THE CODE

```
%Setup
%AddMarkers(gmap__markers,
            Data=markerdata,
            states=1 12 13,
            Var=value,
            lat=lat,
            long=long,
            statevar=state,
            size=0.5,
            color=data,
            datalevels=3,
            colorscheme=green);
title1 .f=swissxb .c=black .h=3 "Example #2";
%CreateMap(US,
           datalevels=3,
           states=1 12 13,
           data=countydata,
           var=xyzy,
           statevar=_state,
           countyvar=_cnty,
           by=county,
           colorscheme=green,
           markers=gmap__markers)
options reset=(title footnote);
%AddImage(image1,
          Image=C:\saslogo.gif)
%CombinePlots(Composite,
             US[0,0,100,100],
             image1[5,5,9,9])
%OutputPlot(Composite,
           dest=C:\Example2.gif)
%Cleanup
```

THE RESULTS

Here is the output from example #2. County lines can now be seen within each state and the SAS logo is visible in the lower left corner. Note the colored, circular marker for the city of Atlanta:

Example #2



USAGE EXAMPLE #3

Now we are going to combine a bit of everything on one page. We will be creating a county level plot of the entire US. Two counties in Florida, two in Georgia and Hawaii will have data. A red marker for Atlanta will be included as well as some overlaid text and the SAS logo. Additional calls to the %CreateMap macro create the plots of Hawaii and Alaska which are then inserted into the lower left corner of the page by the %CombinePlots macro.

THE CODE

```
%Setup
%AddMarkers(gmap__markers,
            Data=markerdata,
            Var=value,
            lat=lat,
            long=long,
            statevar=state,
            size=0.5,
            color=red,
            datalevels=3);
title1 .f=swissxb .c=black .h=3 "Example #3";
footnote1 .f=simplex .c=blue .h=1 "Created By: James L. Ribble";
%CreateMap(US,
            datalevels=3,
            data=countydata,
            var=xyzzy,
            statevar=_state,
            countyvar=_cnty,
            by=county,
            markers=gmap__markers)
options reset=(title footnote);
```

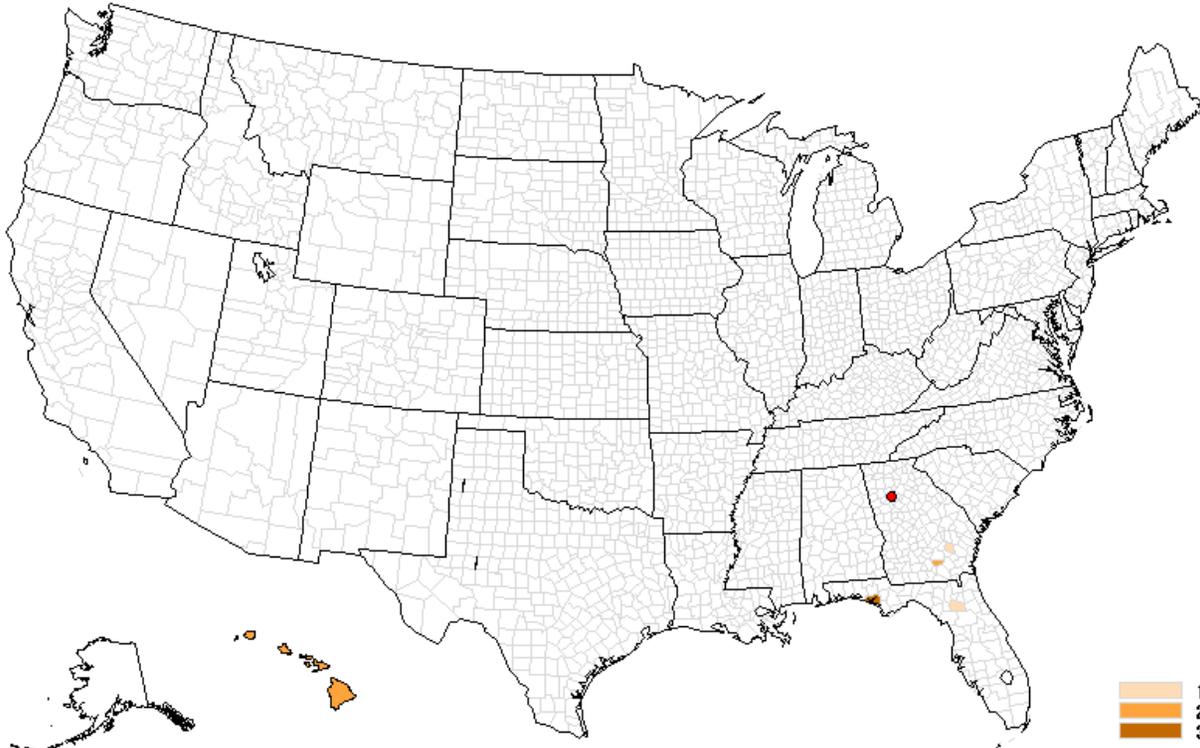
```
%CreateMap(Alaska,
           datalevels=3,
           states=2,
           data=statedata,
           var=xyzzy,
           statevar=_state,
           by=state,
           legend=N)
%CreateMap(Hawaii,
           datalevels=3,
           states=15,
           data=statedata,
           var=xyzzy,
           statevar=_state,
           by=state,
           legend=N)
%AddText(text1,
         text=This text is overlaid,
         x=50,
         y=90,
         color=blue,
         height=3,
         font=swissxb)
%AddImage(image1,
          Image=E:\MyProjects\SGF2008\Paper01\Code\saslogo.gif)
%CombinePlots(Composite,
             US[0,0,100,100],
             Alaska[2,5,17,20]
             Hawaii[20,10,30,20]
             text1[0,0,100,100]
             image1[95,95,99,99])
%OutputPlot(Composite,
           dest=E:\MyProjects\SGF2008\Paper01\Code\Example3.gif)
%Cleanup
```

THE RESULTS

And here is what our final output looks:

Example #3

This text is overlaid



Created By: James L. Ribble

As can be seen all of the text, plots and the logo have been combined into a single graphic image. Creating this required little more knowledge than the parameters needed to call each macro.

CONCLUSION

These macros could be expanded greatly. Virtually any option used by SAS/Graph could be made into a macro parameter allowing the user to supply it at run time without needing to know how to actually code the SAS/Graph logic themselves.

OBTAINING THE CODE

The macros presented in this paper can be obtained by downloading them at SAS.JAMESRIBBLE.COM or by emailing the author at SAS@JAMESRIBBLE.COM.

CONTACT INFORMATION

Your comments and questions are valued. Feel free to contact the author at:

E-mail: SAS@JAMESRIBBLE.COM

Web: SAS.JAMESRIBBLE.COM

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.