

Creating Graphs Using Templates

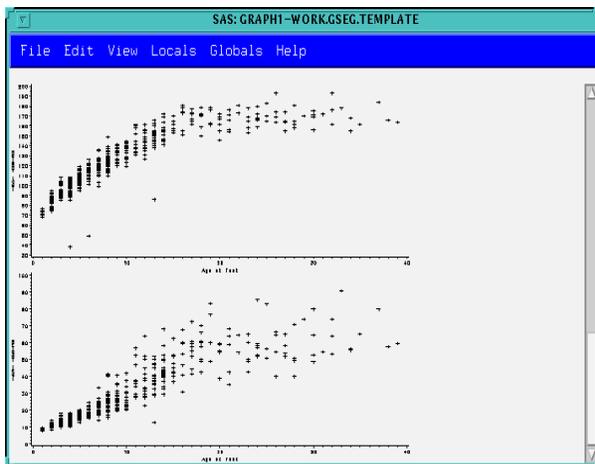
Michelle Gayari, Mayo Clinic, Rochester, MN

Abstract

Sometimes it is an unnecessary waste of paper or inconvenient to print only one graph per page. Templates are a useful tool for displaying more than one graph, either by overlaying graphs or displaying multiple graphs on a page. Templates can also save time since they can be reused as needed once the template catalog is saved. Using SAS/GRAPH® this beginning tutorial will demonstrate how to create templates using the GREPLAY procedure either interactively or in batch mode, how to use templates, and how to store templates once created. GSLIDE will also be shown as a way to display one title for the entire page.

Introduction

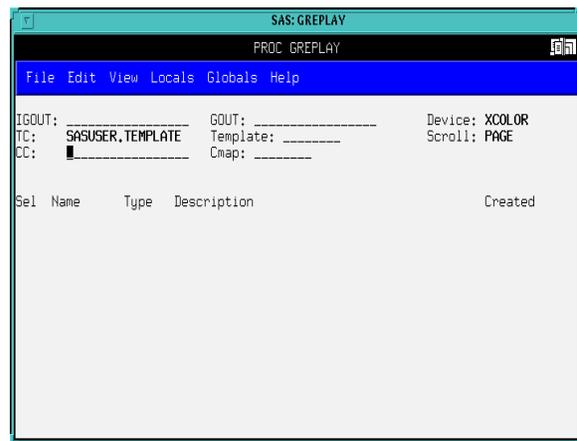
It is useful to display multiple graphs on one page instead of several. For example it may be useful to show a scatterplot of age with height and weight on the same page.



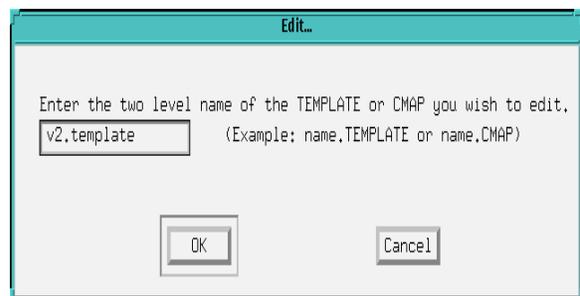
Another method for using templates is to overlay graphs on top of one another. This is useful when using a plot statement that looks like PLOT Y*X=Z since the overlay option cannot be used.

Creating Templates

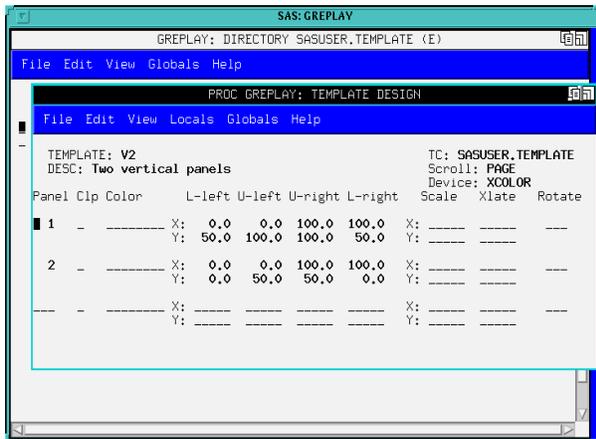
The first step in creating templates is to create a permanent template catalog either through the GREPLAY window or in a program. With windows run a PROC GREPLAY statement. In the TC: area type SASUSER.name of template catalog. For example:



From here, using pmenus, click Locals-->Template Catalog (command line: tc). This will list the templates that are created. So far this screen should be empty. Go to Edit-->Edit. This screen will allow the template to be named. It must be a two level name beginning with the name of the template followed with the word template, for example name.template (command line: edit name.template). For this example the template name will be V2.template to indicate this template will contain two graphs with one on top and one on bottom.



The template is defined by the panel numbers and the coordinates for each panel. The coordinates are based on a 0 to 100 scale and each panel is defined by the upper, lower and left, right bounds of the X and Y axis.



To see the template created, click Locals-->Pre-view template list (command line: preview).



Templates can also be created within a program using the PROC GREPLAY procedure and the TDEF statement. The syntax for the TDEF statement contains the following:

```
TDEF template-name <DES='string'>
      <panel-definition> <...panel-definition-n>;
```

template-name = name of template.

panel-definition = *panel-number*/*<options>*

panel-number = number of the panels being defined.

options = one or more of the following:

CLIP

specifies any panels behind this panel should be clipped.

COLOR=*border-color*

specifies the color of the panel border.

COPY=*panel-number*

specifies the number of the panel definition to be copied to this panel.

DEF

specifies a default panel.

DELETE/DEL

deletes the panel.

DES='string'

specifies a string that appears in the Description field of the catalog entry for the template.

LLX=*x*

specifies the X coordinate for the lower-left corner of the panel.

LLY=*y*

specifies the Y coordinate for the lower-left corner of the panel.

LRX=*x*

specifies the X coordinate for the lower-right corner of the panel.

LRY=*y*

specifies the Y coordinate for the lower-right corner of the panel.

ROTATE=*degrees*

specifies the rotation angle for the panel.

SCALEX=*factor*

specifies the scale factor for the X coordinates in the panel.

SCALEY=*factor*

specifies the scale factor for the Y coordinates in the panel.

ULX=*x*
 specifies the X coordinate for the upper-left corner of the panel.

ULY=*y*
 specifies the Y coordinate for the upper-left corner of the panel.

URX=*x*
 specifies the X coordinate for the upper-right corner of the panel.

URY=*y*
 specifies the Y coordinate for the upper-right corner of the panel.

XLATEX=*distance*
 specifies the distance to move the X coordinates of the panel.

YLATEX=*distance*
 specifies the distance to move the Y coordinates of the panel.

The following is an example creating the same template as before with two vertical graphs.

```
PROC GREPLAY TC=SASUSER.TEMPLATE
  NOFS;
```

```
/* defines template */
TDEF v2 DES='Two vertical squares'
```

```
/* define panel 1 */
```

```
1/lx=0 lly=50
  ulx=0 uly=100
  urx=100 ury=100
  lrx=100 lry=50
```

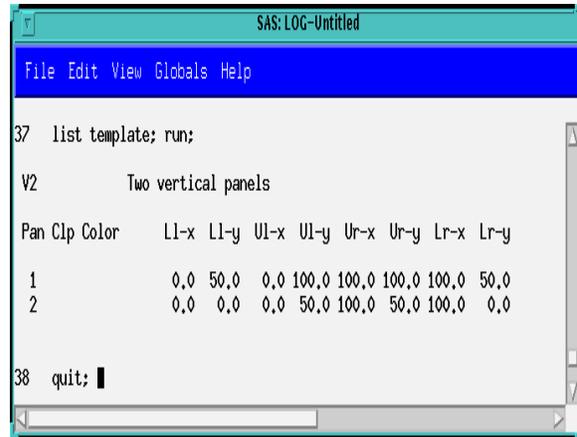
```
/* define panel 2 */
```

```
2/lx=0 lly=0
  ulx=0 uly=50
  urx=100 ury=50
  lrx=100 lry=0;
```

```
/* assign current template */
TEMPLATE V2;
```

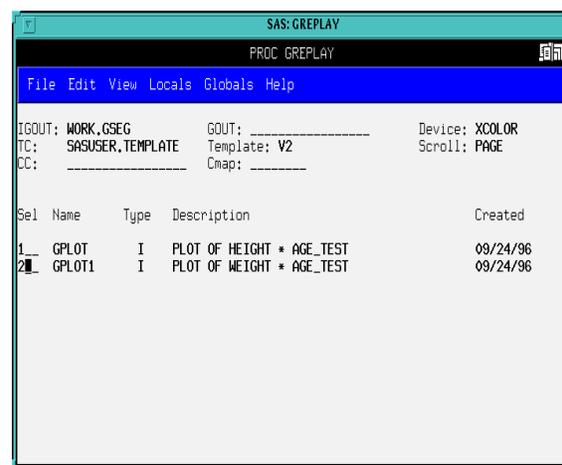
```
/* list contents of current template */
LIST TEMPLATE;
QUIT;
```

The list template statement displays the following in the sas log:



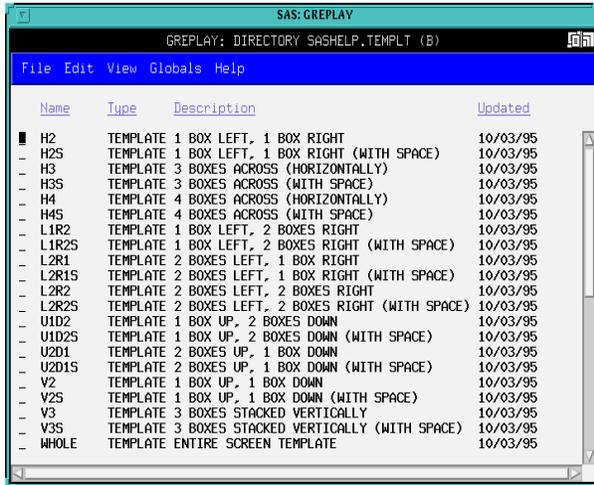
Using Templates

Once the template is created then each graph needs to be assigned a panel number corresponding to where each graph should be displayed. Also the name of the template used needs to be filled in on the template line.



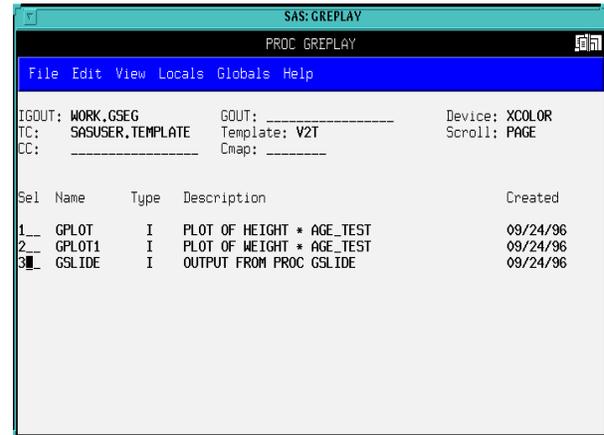
The graphs will automatically be put in the panels defined earlier.

There are some predefined templates that can be used and are located in SASHELP.TEMPLT.



```
PROC GSLIDE GOUT=WORK.GSEG;
  TITLE1 'Example of One Title for Entire
  Page Using Gslide';
RUN; QUIT;
```

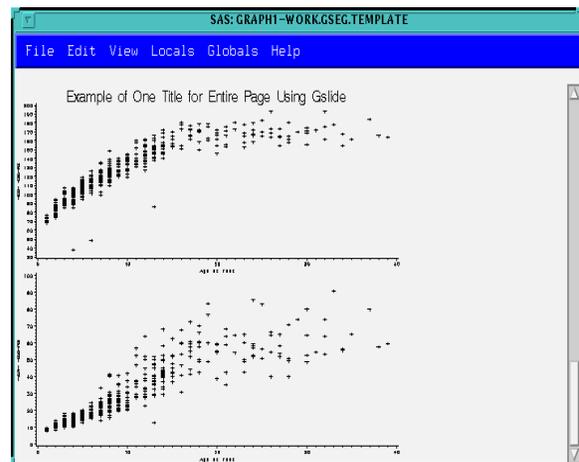
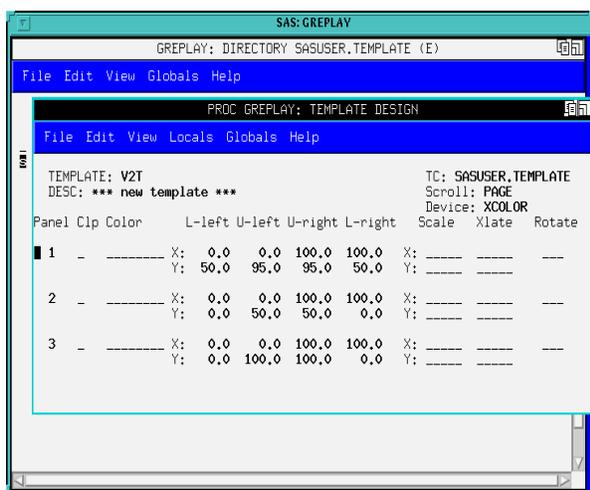
Next the graph can be created using the template V2T and defining the gslide title as panel 3.



GSLIDE

Gslide can be used to display one title for the entire page. First create a template with one panel that consists of the entire page. The other panels may need to be adjusted in order to make room for the title. For example, create a template similar to the one created previously but lower the upper bounds for panel 1 and include a panel 3 which includes the entire page. This template will be called V2T.

This will produce the following output:



The next step is to run a PROC GSLIDE statement to define the gslide. In this example the gslide will be defined to the work library.

Summary

Templates are a useful tool for displaying multiple graphs. They can either be created through windows or in a program with many options.

References

SAS Institute Inc., *SAS/GRAPH Software, Volume 1, Reference, Version 6, First Edition*, Cary, NC: SAS Institute, Inc., 1990

SAS Institute Inc., *SAS/GRAPH Software, Volume 2, Reference, Version 6, First Edition*, Cary, NC: SAS Institute, Inc., 1990

Acknowledgments

SAS and SAS/GRAPH are registered trademarks of sas Institute Inc. in the USA and other countries.

® indicates USA registration.

Author

Michelle Gayari
Mayo Clinic
Biostatistics
200 1st Street SW
Rochester, MN 55905
507/284-3710
gayari@mayo.edu