

Paper CC03

Making Presentations More Fun with DATA Step Graphics Interface (DSGI)

Hui-Ping Chen, Eli Lilly and Company, Indianapolis, Indiana

ABSTRACT

Microsoft PowerPoint is powerful and most popular software for presentation. However, it is more convenient to present your analyzed data from SAS® Output Window or Result Viewer Window directly when you have already populated all information by SAS. The advantages of showing the results from SAS directly are 1) reducing mistakes when manually copy or re-type the results into another software, such as PowerPoint; 2) it is easy to re-fresh the results when the data is updated. So, you can use GSLIDE procedure or DSGI data step to create slides or graphs for the presentation based on the results you get from previous procedures (PROC). However, it is more efficient, more fun, and much easier to create slides with DSGI, especially to create drill-down graphs.

INTRODUCTION

The advantages of presenting the analyzed data from SAS window directly are reducing re-works and ensuring accuracy. Using DSGI to display the numeric results, texts, or graphs, respectively, or combine texts with graphs or several graphs into one page for the presentation is very efficient.

DSGI has 5 key functions. They are

- 1) GINIT, which is initializing DSGI in order to activate it;
- 2) GRAPH, which is opening a graphic segment;
- 3) GSET, which is setting up attributes for an entire graph;
- 4) GDRAW, which is generating customized graphics;
- 5) GTERM, which is ending DSGI.

Function GDRAW enables you to create texts or draw graphs. Function GRAPH enables you to 1) insert existing graphs created by previous procedures, such as PROC GPLOT or PROC GCHART; 2) import existing graphs created by different SAS programs or ordinary photos in BMP or JPEG format. Function GSET enables you to link a website or other HTML files. That is, you can create Drill-Down graphs with function GSET and it can make your presentation more dynamic.

In other words, DSGI is not only able to create a customized graph, but can also call existing graphs generated by other SAS procedures (PROC). DSGI can also display multiple graphics in one output with common titles or footnotes or, add Texts or symbols to the graphics.

CREATE A SIMPLE GRAPH WITH DSGI

Here is an example to create graphs with DSGI.

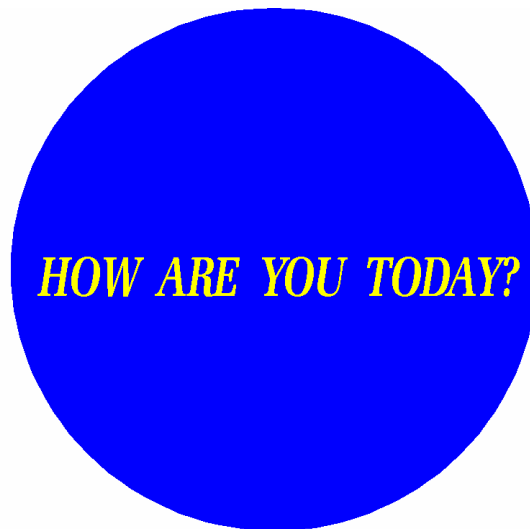
```
DATA _NULL_;
  /*Initialize DSGI*/
  RC = GINIT ();
  /*Open a graphic segment*/
  RC = GRAPH ('CLEAR');

  /*Assign color, Yellow, to 1*/
  RC = GSET ('COLREP', 1, 'YELLOW');
  /*Assign color, Blue, to 2*/
  RC = GSET ('COLREP', 2, 'BLUE');
  /*Fill-in area with No. 2 color, Blue*/
  RC = GSET ('FILCOLOR', 2);
  /*Fill-in type is Solid*/
  RC = GSET ('FILTYPE', 'SOLID');
  /*Draw a Solid Blue circle, from 0 to 360 degree at coordinate (72, 50) with
  semi-diameter = 50*/
  RC = GDRAW ('PIE', 72, 50, 50, 0, 360);
```

```

/*Set up text color as Yellow*/
RC = GSET ('TEXCOLOR', 1);
/*Set up text height is 10 unit*/
RC = GSET ('TEXHEIGHT', 10);
/*Set up text font as Swiss Bold*/
RC = GSET('TEXTFONT', 'ZAPFBI');
/*Write down text, "HOW ARE YOU TODAY?" starting from coordinate, (21, 45)
*/
RC = GDRAW ('TEXT', 21, 45, 'HOW ARE YOU TODAY?');
/*Close the graphics segment*/
RC = GRAPH ('UPDATE');
/*Terminate DSGI*/
RC = GTERM ();
RUN; /*Display output*/

```



You can

- 1) Draw ARC, BAR, ELLARC, ELLIPSE, LINE, MARK, PIE, or [TEXT](#) with GDRAW function.
- 2) Import external photos into SAS with GDRAW function, [IMAGE](#).
- 3) Insert existing graphs (with NAME = option) with GRAPH function, [INSERT](#).
- 4) Display existing graphs (with NAME = option) with GRAPH function, [PLAY](#). It is similar to the function of GREPLAY.
- 5) Specify a valid URL or file path to invoke with GSET function, [HTML](#). So when you click the graphic element in a web page, it will automatically link to the next graph.
- 6) Use FILENAME and GSFNAME (GOPTION statement) to output graph as an external graphic file while working with DSGI.

IMPORT EXTERNAL PHOTOS INTO SAS

You can import graphic files with PROC GIMPORT. However, this procedure only supports the external graphic files with CGM format. Therefore, you can use PROC GSLIDE, ANNOTATE facility, or DSGI to import the external graphic files into SAS with more flexibility. For displaying images with those 3 methods mentioned above in your graphs, the most popular image file types supported by SAS/GRAPH are BMP, GIF (Graphics Interchange Format), JPEG, and TIFF (Tagged Image File Format).

Here is an example with GSLIDE procedure.

```

FILENAME GETGRAPH 'external-image-file';
PROC GSLIDE IFRAME = GETGRAPH IMAGESTYLE = FIT;
  NOTE MOVE = (25, 45) HEIGHT = 2 FONT = 'SWISS'
    'IMPORT EXTERNAL PHOTO WITH GSLIDE';

```

```
RUN;  
QUIT;
```

Not only can you import an external image file into SAS, but also add some texts on the image file with NOTE statement. TILE is another option for IMAGESTYLE, which means that the image is copied as many times as needed to fill the area, while FIT means to stretch the image so that a single image fits within the area.

Here is an example with ANNOTATE facility.

```
DATA ANNO;  
LENGTH FUNCTION $5.;  
RETAIN FUNCTION 'MOVE' FUNCTION 'IMAGE' XSYS '3' YSYS '3';  
  
X = 0; Y = 0; FUNCTION = 'MOVE';  
OUTPUT;  
X = 50; Y = 50; IMGPATH = 'external-image-file';  
STYLE = 'FIT';  
FUNCTION = 'IMAGE';  
OUTPUT;  
  
X = 0; Y = 50; FUNCTION = 'MOVE';  
OUTPUT;  
X = 50; Y = 100; IMGPATH = 'external-image-file';  
STYLE = 'FIT';  
FUNCTION = 'IMAGE';  
OUTPUT;  
  
X = 50; Y = 50; FUNCTION = 'MOVE';  
OUTPUT;  
X = 100; Y = 100; IMGPATH = 'external-image-file';  
STYLE = 'FIT';  
FUNCTION = 'IMAGE';  
OUTPUT;  
  
X = 50; Y = 50; FUNCTION = 'MOVE';  
OUTPUT;  
X = 100; Y = 0; IMGPATH = 'external-image-file';  
STYLE = 'FIT';  
FUNCTION = 'IMAGE';  
OUTPUT;  
RUN;  
  
PROC GSLIDE ANNOTATE = ANNO;  
RUN;  
QUIT;
```

The codes above draw 4 images from (0, 0) to (50, 50), (0, 50) to (50, 100), (50, 50) to (100, 100), and (50, 0) to (100, 50). If you do not use MOVE function for displaying one single image, you can just set up the upper right coordinates, such as (100, 100), and the image will be displayed within the range from (0, 0) to (100, 100). While using ANNOTATE to display an image, you need to use PROC GSLIDE to output ANNOTATE dataset. If you like to add some texts on the images, PROC GSLIDE with NOTE statement will be very helpful.

Here is an example with DSGI.

```
DATA _NULL_;  
RC = GINIT();  
RC = GRAPH('CLEAR');  
RC = GDRAW('IMAGE', "external-image-file", x1, y1, x2, y2, 'FIT');  
RC = GRAPH('UPDATE');
```

```
RC = GTERM();
RUN;
```

This image will be fitted within the range from (x1, y1) to (x2, y2). Same as the function of Annotate facility, DSGI allows you to import several external image files into SAS. If you like to add some texts on the image, you can use GDRAW function with TEXT element or just use TITLE / FOOTNOTE statements, same as the function of GSLIDE. However, you cannot use ANNOTATE facility to display external image without GSLIDE procedure.

CREATE DRILL-DOWN GRAPHS WITH DSGI

DSGI has drill-down function to support linking function. The drill-down function will provide you a dynamic way while generating your presentation slides. It is much easier to create drill-down graphs with DSGI than writing HTML codes.

Here is an example to create a drill-down graph with DSGI.

```
%LET PATH = C:\;
FILENAME PATH "&PATH";
ODS HTML BODY = 'BODY.HTML' PATH = PATH; (1)

DATA _NULL_;
  RC = GINIT ();
  RC = GRAPH ('CLEAR');

  RC = GSET ('HTML', " HREF = ""IMAGE.HTML"" "); (2)
  RC = GDRAW ('BAR', x1, y1, x2, y2); (3)
  RC = GRAPH ('INSERT', 'name-entry'); (4)

  RC = GRAPH ('UPDATE');
  RC = GTERM ();
RUN;

ODS HTML CLOSE;
ODS LISTING;
QUIT;
```

As mentioned above, you can set up a HTML string by using GSET function with HTML element. So when you click the string, it will automatically create a link to the next graph. In this example shown above,

- 1) You need to open HTML destination with ODS HTML statement;
- 2) The HTML string must be **HREF =** followed by any file name with extension, HTML, such as
 - RC = GSET('HTML', ' HREF = "IMAGE.HTML" ');

If the macro variable, &PATH, is included in the string, you need to use double quotation mark instead of single quotation mark, such as

- RC = GSET ('HTML', " HREF = ""&PATH\IMAGE.HTML"" ");

- 3) Function GDRAW is to create a hot zone.
- 4) Function GRAPH with element, INSERT, allows you to insert an existing graph created from previous procedure, such as PROC GPLOT, with NAME = 'name-entry'. The hot zone in 3) will activate the link function. That is, when you click on the inserted graph, it will automatically link to the next graph and be written into BODY =. Not only can you insert an existing graph, you also can import any BMP or JPEG images into SAS with a link function.

It is very easy to generate the slides for presentation by creating HTML links. During the presentation, you just need to click the slide and it will link to the next slide as a loop.

Here is an example to create slides with the link function.

```
GOPTION RESET = ALL DEVICE = HTML CBACK = YELLOW;
```

```

%LET PATH = C:\SAS;
FILENAME PATH "&PATH";

ODS HTML BODY = 'FIRST.HTML' PATH = PATH;
DATA _NULL_;
  RC = GINIT();
  RC = GRAPH('CLEAR');
  RC = GSET('HTML', 'HREF = "ZERO.HTML"');
  RC = GDRAW('BAR', 0, 0, 150, 120);
  RC = GSET('TEXHEIGHT', 10);
  RC = GSET('TEXTFONT', 'ZAPFBI');
  RC = GDRAW('TEXT', 5, 55, "DSGI with drill down function");
  RC = GRAPH('UPDATE');
  RC = GTERM();
RUN;

ODS HTML BODY = 'SECOND.HTML' PATH = PATH;
DATA _NULL_;
  RC = GINIT();
  RC = GRAPH('CLEAR');
  RC = GSET('HTML', 'HREF = "FIRST.HTML" ');
  RC = GSET('TEXHEIGHT', 12);
  RC = GSET('TEXTFONT', 'ZAPFBI');
  RC = GDRAW('TEXT', 15, 55, 'It is just like a loop');
  RC = GRAPH('UPDATE');
  RC = GTERM();
RUN;

ODS HTML BODY = 'ZERO.HTML' PATH = PATH;
DATA _NULL_;
  RC = GINIT();
  RC = GRAPH('CLEAR');
  RC = GSET('HTML', 'HREF = "SECOND.HTML" ');
  RC = GDRAW('BAR', 0, 0, 150, 150);
  RC = GDRAW('IMAGE', 'external-image-file', 0, 0, 140, 100, 'FIT');
  RC = GSET('TEXHEIGHT', 10);
  RC = GSET('TEXTFONT', 'ZAPFBI');
  RC = GDRAW('TEXT', 10, 90, "Importing an existing picture");
  RC = GRAPH('UPDATE');
  RC = GTERM();
RUN;
ODS HTML CLOSE;
QUIT;

```

CONCLUSION

When making a presentation dealing with statistical information, we usually copy from existing statistical or descriptive results and paste into PowerPoint or insert the statistical graphs into PowerPoint. However, generating slides while running SAS program, it can avoid re-work and ensure the accuracy of information. That is, SAS is not just statistical analyzing software, but also a software full of fun and challenge.

REFERENCES

SAS® Institute (1999), SAS/GRAPH® Software: Reference, Version 8,
 Copyright 1999 by SAS® Institute Inc., Cary, NC, USA
 Paper: 049-30 (Creating Flexible Graphics with Data step Graphics Interface (DSGI))

CONTACT INFORMATION

Your comments and questions are valuable and appreciated. Author can be reached at

Hui-Ping Chen
 Eli Lilly and Company
 Lilly Corporate Center
 Indianapolis, IN 46285 U.S.A.

Work Number: (317) 651 5336

Email: huiping@lilly.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.