Paper 216-26

Your Graphs on the Web with SAS/GRAPH® Version 8

Francesca Pierri, C.A.S.I. - Università di Perugia, Perugia, Italy

ABSTRACT

The new features of SAS/GRAPH and the Output Delivery System provide the possibility of creating interesting graphs on the Web even lacking an extensive knowledge about the Hyper Text Markup Language (HTML). In particular the new enhancements allow to create *drill-down* graphs and define a link connecting each clickable area with another output, such as a report or another more detailed graph; moreover there is the chance of organizing the web page into areas and displaying for instance a Table of Contents and a graph in a single frame.

This paper shows a few graphs and programs written with the aim of drawing and viewing them on the web. In particular it highlights new Version 8 features for the GMAP procedure, the possibility of creating *drill-down* graphs, combining output from different procedures and defining custom layouts.

The coding practices discussed in this article are aimed at users with average to advanced SAS/GRAPH experience; the products used are SAS/Base and SAS/GRAPH, with no limitations in operating systems.

INTRODUCTION

The use of graphs to display information is really incisive, because a graph has the capability to attract attention and give an immediate idea of the distribution of the data. Moreover, if you like to show the results of your job or research to other people in the group, you could think of publishing these data on an intranet. So you can say that graphs and Web are surely a winning couple.

With SAS/GRAPH software Version 6.12 you were able to produce graphs utilizing different output devices and could include them on your reports, articles, web pages, but with the latter you were supposed to write HTML code. In particular the GMAP procedure allowed you to show data on regional maps, use the ANNOTATE= option in writing some interesting things on them, and you could say that it was a good procedure, but it would have been preferable having the choice of clicking an area of the graph and show more detailed information.

Version 8 of SAS/GRAPH software has implemented the GMAP, CGHART and GPLOT procedures with two options (HTML= , HTML_LEGEND=) which use the HTML functionality to create graphics output with *drill-down* capabilities. So, now, you can very easily publish your graphs on the Web and click on one of its regions to see something else.

The next sections of this paper focus on different ways of creating graphics output for the Web with SAS/GRAPH software and show examples of how to do it.

SAS/GRAPH ON THE WEB

SAS/GRAPH software Version 8 gives three different ways to produce output on the Web:

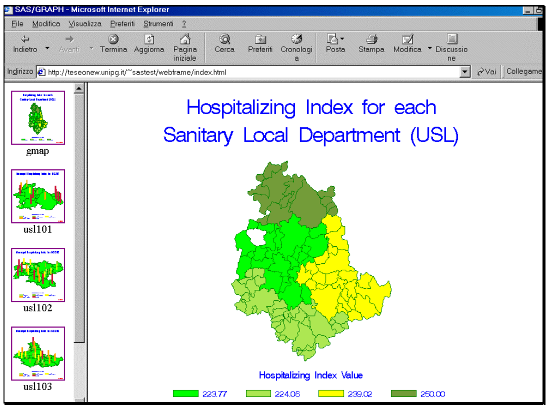
1. SAS/GRAPH Web drivers (GIF, HTML, WEBFRAME):
   - the GIF driver creates GIF format files for graphs and if you like to publish graphs on the Web you have to reference them on an HTML file;
   - the HTML and WEBFRAME drivers generate both the HTML and the GIF files: all you have to do is specifying the driver and SAS does the rest. In particular the WEBFRAME driver generates *thumbnail-size* graphs you can click to see the full-size graphs.
2. Output Delivery System (ODS). If you use ODS and GIF, or JAVA or ACTIVEX drivers, SAS automatically generates the HTML file needed to display the graphs. With ODS you name the HTML file and have more control on the Web page you create. For example you can combine graphics and non-graphics output, generate a table of contents linking each piece of output, create *drill-down* graphs.
3. Output data set of SAS/GRAPH (image map). You can use this approach if you have the need of displaying *drill-down* graphs and customize your Web page. In this case you use the IMAGEMAP= procedure option and a SAS/GRAPH macro to create an image map of your *drill-down* graph and then you have to create your HTML file and write the HTML tags to display the output, associate the image map with it and resolve the image map links.

HTML AND WEBFRAME DEVICE DRIVERS

The program you write if you want to use the HTML or the WEBFRAME driver is the same, you only change the name of the driver after the goptions DEVICE= option; the results, however, are slightly different.

If you specify DEVICE=HTML you obtain an *index.html* file and a GIF file for every graph you have specified on your program. When you view the *index.html* file on your browser, you see all the graphs in sequence on a single web page. You can't change the name of the html file, so, if you run more than one program specifying GOPTIONS DEVICE=HTML, you need to rename the html file on your computer or change the output location every time.

The WEBFRAME device driver defines two frames for displaying the output, so, in order to display the output on the web, you need a browser supporting HTML frames. The layout of the Web page is broken down into two views: on the left side you can see the thumbnail-size graphs and on the right the current full size graph.


SAS/GRAPH - Microsoft Internet Explorer
File Modifica Visualizza Preferiti Strumenti ?
Indietro Avanti Termina Aggiorna Pagina iniziale Cerca Preferiti Cronologia Posta Stampa Modifica Discussione
Indirizzo http://teseonew.unipg.it/~sastest/webframe/index.html
Vai Collegame
gmap
usl101
usl102
usl103
Hospitalizing Index for each Sanitary Local Department (USL)
Hospitalizing Index Value
223.77
224.06
239.02
250.00

Picture 1

The WEBFRAME device generates:

- an *index.html* file, as previously described;
- a *sasthumb.html*, referencing all the GIF images created by the WEBFRAME driver and linking each thumbnail to its corresponding full size graph;
- *<graphname>.html* files, one for each file produced by the procedure, displaying only the full size graph mentioned;
- *<GRSEG entry name>.gif*, one for each graph, and one for each thumbnail size graph, with the difference in name: the latter has prefix *"f"* before the name defined in the GRSEG entry.

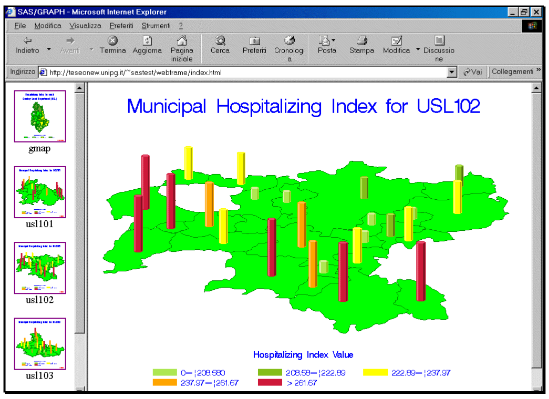
If you specify a BY statement on the graphic procedures, it is better to use the GOPTION NODISPLAY option before generating graphs, to prevent each graph from overwriting the *index.html* file. When your program has stopped you can view the results running the GREPLAY procedure.

Example 1 creates graphs shown on Picture 1 and Picture 2.

Example 1

/* define location for output file */
```
   filename out '/u1/sastest/WWW/webframe';
```

```
goptions reset=all nodisplay device=webframe
         gsfname=out gsfmode=replace
         gunit=pct cback=white ftext=swiss
         htitle=6 htext=2.8;
```


SAS/GRAPH - Microsoft Internet Explorer
http://teseonew.unipg.it/~sastest/webframe/index.html
gmap
usl101
usl102
usl103
Municipal Hospitalizing Index for USL102
Hospitalizing Index Value
0—|208.580
208.58—|222.89
222.89—|237.97
237.97—|261.67
> 261.67

Picture 2

/* define title and footnote statements for the Umbria map */

```
   title1 c=blue
   "Hospitalizing Index for each";
   title2 h=6 c=blue
   "Sanitary Local Department(USL)";
   footnote h=3 j=r "REGIONMAP";
```

/* define pattern characteristics */

```
   pattern1 v=solid c=lime ;
   pattern2 v=solid c=bigy;
   pattern3 v=solid c=vigy;
   pattern4 v=solid c=degy;
```

/* modify the default legend with the legend1 statement */

```
   legend1 label =(c=blue position=(top center)
                    "Hospitalizing Index Value")
           value=(c=blue f=swiss h=2.5);
```

/*
dat.drillusl data set contains the index variable for each Sanitary Local Department (USL); *map.umbria* data set is the regional map data set.
*/

```
   proc gmap data=dat.drillusl map=map.umbria;
        id usl;
        choro index/ discrete coutline=green
                     legend=legend1;
   run;
```

/* define title and footnote statements for the USL maps */

```
   option nobyline;
   title1 "Municipal Hospitalizing Index for
   USL#byval(usl)";
   footnote h=3 j=r "USLMAP";
   pattern1 v=ms c=lime ;
   pattern2 v=solid c=bigy;
   pattern3 v=solid c=vigy;
   pattern4 v=solid c=yellow;
   pattern5 v=solid c=orange;
   pattern6 v=solid c=vipk;
   legend1 label=(c=blue position=(top center)
                  "Hospitalizing Index Value")
                   value=(c=blue f=swiss h=2.5);
```

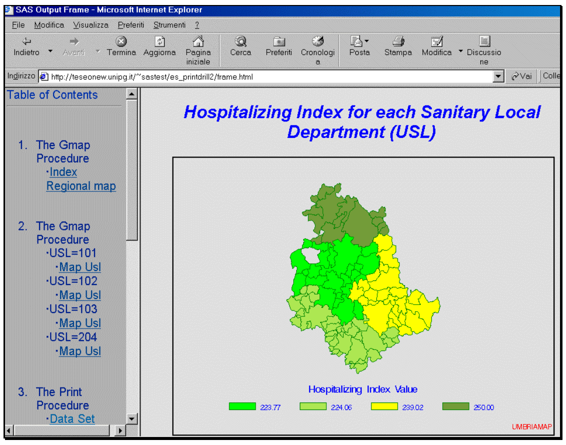
/*
sort data sets depending on usl variable; *dat.indice* data set contains the index variable for each Commune.
*/

```
   proc sort data=dat.indice ;
        by usl;
   proc sort data=map.umbria out=map.umbsort;
        by usl;
   proc gmap data=dat.indice map=map.umbsort;
        by usl;
```

```
        id comres;
        block indpct / coutline=green
                       cblkout=same
                       legend=legend1
                       shape=hexagon
                       name="usl101";
   run;
   quit;
   goptions display dev=webframe;
   proc greplay igout=work.gseg nofs;
   replay _all_;
   run;
   quit;
```

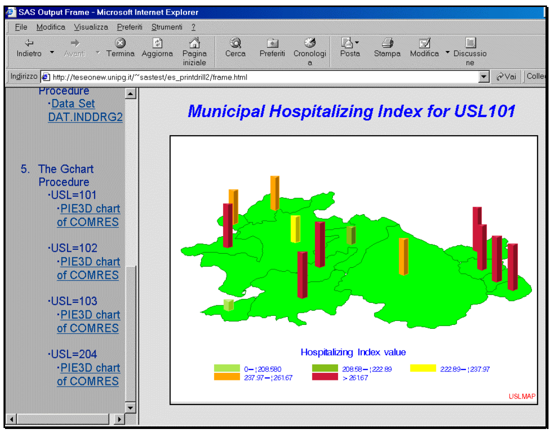
ODS AND GIF DEVICE DRIVER

Using ODS with SAS/GRAPH software gives you the same possibility of generating Web output with HTML and WEBFRAME drivers, but, if you just know some HTML language, you can combine the output from different procedures, recall the output with the *drill-down* facility, generate a Table of Contents to link to the output and so on.



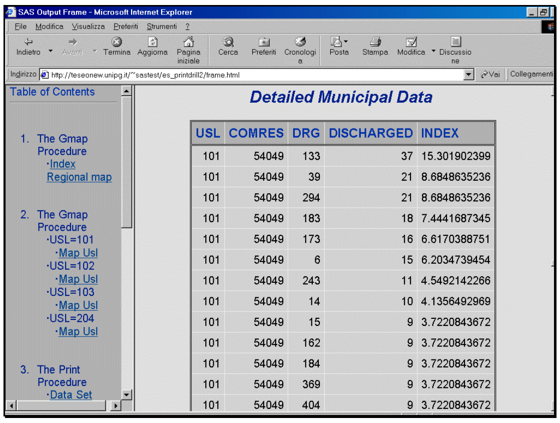
Picture 3

The output you see on Picture 3 is obtained using the ODS method; the frame is divided into two sides: on the left there is a Table of Contents and on the right the Regional Umbria Map. There are four sub-regions (Sanitary Local Departments), each one is coloured in a different way, depending on the acquired value of the index variable.



Picture 4

If you click on a sub-region the view on the right changes, showing the map of the sub-region divided into Communes (Picture 4). On each Commune there is a tower meant to show a report with detailed data concerning that Commune (Picture 5).

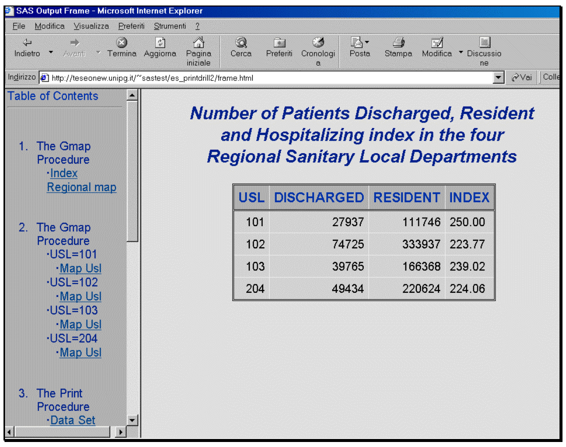

SAS Output Frame - Microsoft Internet Explorer
File Modifica Visualizza Preferiti Strumenti ?
Indietro Avanti Termina Aggiorna Pagina iniziale Cerca Preferiti Cronologia Posta Stampa Modifica Discussione
Indirizzo http://teseonew.unipg.it/~sastest/es_printdrill2/frame.html
Vai Collegamenti
Table of Contents
1. The Gmap Procedure
·Index Regional map
2. The Gmap Procedure
·USL=101
·Map Usl
·USL=102
·Map Usl
·USL=103
·Map Usl
·USL=204
·Map Usl
3. The Print Procedure
·Data Set

Detailed Municipal Data

USL	COMRES	DRG	DISCHARGED	INDEX
101	54049	133	37	15.301902399
101	54049	39	21	8.6848635236
101	54049	294	21	8.6848635236
101	54049	183	18	7.4441687345
101	54049	173	16	6.6170388751
101	54049	6	15	6.2034739454
101	54049	243	11	4.5492142266
101	54049	14	10	4.1356492969
101	54049	15	9	3.7220843672
101	54049	162	9	3.7220843672
101	54049	184	9	3.7220843672
101	54049	369	9	3.7220843672
101	54049	404	9	3.7220843672

Picture 5

You can get information from the UMBRIAMAP and USLMAP legends, too: clicking on the former, you get the result of PRINT procedure (Picture 6), clicking on the latter, a pie chart will appear showing percentage of patients discharged from each Sanitary Local Department (Picture 7).


SAS Output Frame - Microsoft Internet Explorer
File Modifica Visualizza Preferiti Strumenti ?
Indietro Avanti Termina Aggiorna Pagina iniziale Cerca Preferiti Cronologia Posta Stampa Modifica Discussione
Indirizzo http://teseonew.unipg.it/~sastest/es_printdrill2/frame.html
Vai
Table of Contents
1. The Gmap Procedure
·Index Regional map
2. The Gmap Procedure
·USL=101
·Map Usl
·USL=102
·Map Usl
·USL=103
·Map Usl
·USL=204
·Map Usl
3. The Print Procedure
·Data Set

Number of Patients Discharged, Resident and Hospitalizing index in the four Regional Sanitary Local Departments

USL	DISCHARGED	RESIDENT	INDEX
101	27937	111746	250.00
102	74725	333937	223.77
103	39765	166368	239.02
204	49434	220624	224.06

Picture 6

As you can see from the example code, the following things are to be specified:

- *HTML= option* to make the *drill-down* available on the map
- *HTML_LEGEND=* option to activate legend *drill-down*
- *links* connecting each clickable area with the corresponding graph or report. These links are HTML commands stored in a new variable of the data set defined as input of the procedure. When the ODS HTML statement creates the body file, it includes these commands in it.
- *ods html statement,* one for each procedure. At least one *'body'* file is required.

The following code (Example 2) produces the output of Pictures 3 to 7, several comments between the code explain the meaning of the most important statements and options.

Example 2

/* define locations for output files */

```
   filename odsout '/u1/sastest/WWW/drill_down';
```

/*

start sending output to html instead of graph and output window

*/

```
   ods listing close;
```

/* define graphic options and device driver */

```
   goptions reset=global gunit=pct cback=white
            ftext=swiss htitle=4 htext=3.5
            device=gif transparency border
            xpixels=650 ypixels=470;
```

/*

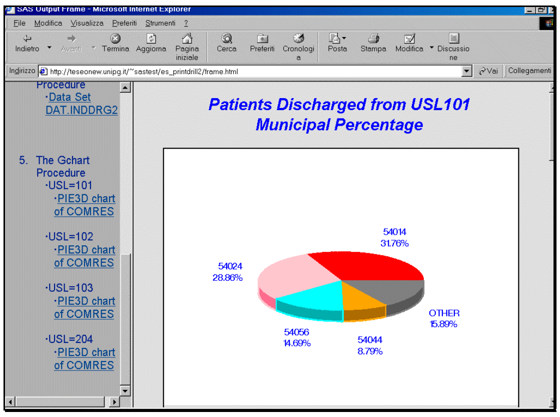
create the new variable *drilleg* and *usldrill* to define a link for each clickable area. The form is:

HREF="name.html <*anchor name*>"

<*anchor name*> must be a unique name and it must match the name of the corresponding output

*/

```
   data dat.drillusl;
        length drilleg usldrill $40.;
        set dat.drillusl;
        drilleg='HREF="legbody.html#leg"';
   if usl=101 then
           usldrill='HREF="uslbody.html#usl1"';
   else if usl=102 then
           usldrill='HREF="uslbody.html#usl2"';
   else if usl=103 then
           usldrill='HREF="uslbody.html#usl3"';
   else if usl=204 then
           usldrill='HREF="uslbody.html#usl4"';
```


SAS Output Frame - Microsoft Internet Explorer
File Modifica Visualizza Preferiti Strumenti ?
Indietro Avanti Termina Aggiorna Pagina iniziale Cerca Preferiti Cronologia Posta Stampa Modifica Discussione
Indirizzo http://teseonew.unipg.it/~sastest/es_printdrill2/frame.html
Vai Collegamenti
Procedure
·Data Set DAT.INDDRG2
5. The Gchart Procedure
·USL=101
·PIE3D chart of COMRES
·USL=102
·PIE3D chart of COMRES
·USL=103
·PIE3D chart of COMRES
·USL=204
·PIE3D chart of COMRES
Patients Discharged from USL101 Municipal Percentage
54014 31.76%
54024 28.86%
OTHER 15.89%
54056 14.69%
54044 8.79%

Picture 7

/*

define the following:

- *mapbody.html* file referencing the graphics output of GMAP procedure;
- *contents.html* file containing HTML information regarding the Table of Contents displayed on the left Web page side;
- *frame.html* file, the main file you view on the web, once the graphs are created.
- *path*, the destination for the graphic output generated by the ODS HTML statement.

*/

```
   ods html body="mapbody.html"
            contents="contents.html"
            frame="frame.html"
            path=odsout nogtitle;
```

/* define title, footnote and patterns */

```
   title1 c=blue "Hospitalizing Index for each
   Sanitary Local Department (USL)";
   footnote h=2.5 j=r c=red "UMBRIAMAP";
   pattern1 v=solid c=lime ;
   pattern2 v=solid c=bigy;
   pattern3 v=solid c=vigy;
   pattern4 v=solid c=degy;
```

/* modify the default legend with the legend1 statement */

```
   legend1 label=(c=blue position=(top center)
                  "Hospitalizing Index Value")
           value=(c=blue h=2.5);
```

/*

create the Umbria Map. The SAS data set is the same previously defined in Example 1; *HTML=usldrill* allows controlling the link of each sub-regional area; *HTML_LEGEND=drilleg* allows controlling the link for the legend.

*/

```
   proc gmap data=dat.drillusl map=map.umbria;
        id usl;
        choro index/ discrete
                     coutline=green
                     html=usldrill
                     legend=legend1
                     html_legend=drilleg
                     des="Index Regional map"
                     name="UMBRIA";
   run;
   quit;
```

/*
define a new body html file (*uslbody.html*) for the four graphics output concerning the sub-region areas; *anchor=usl1* defines the anchor name. This name is unique and is the same used on the HTML HREF statement above. This allows to display the right output clicking on each sub-region area. The BY statement in PROC GMAP generates four graphs, one for each USL; the anchor name usl1 is automatically incremented in usl2 usl3 and usl4
*/

```
   ods html body="uslbody.html"
            anchor="usl1"
            path=odsout;
   goptions notransparency border;
   proc sort data=map.umbria out=map.umbsort;
        by usl;
   run;
   proc sort data=dat.indice;
        by usl;
   run;
   goptions nobyline ;
```

/* define title, footnote and patterns */

```
   title1 c=blue "Municipal Hospitalizing Index
   for USL#byval(usl)";
   footnote h=2.5 j=r c=red "USLMAP";
   pattern1 v=ms c=lime ;
   pattern2 v=solid c=bigy;
   pattern3 v=solid c=vigy;
   pattern4 v=solid c=yellow;
   pattern5 v=solid c=orange;
   pattern6 v=solid c=vipk;
   legend1 label=(c=blue position=(top center)
                 "Hospitalizing Index Value")
           value=(c=blue h=2.5);
```

/*
create the new variable *drilleg2* to define a link for the USLMAP legend.
*/

```
   data dat.indice;
        set dat.indice;
        if usl=101 then
           drilleg2='HREF="legbody2.html#usl1"';
        else if usl=102 then
           drilleg2='HREF="legbody2.html#usl2"';
        else if usl=103 then
           drilleg2='HREF="legbody2.html#usl3"';
        else if usl=204 then
           drilleg2='HREF="legbody2.html#usl4"';
   run;
```

/*
create a graph for each Sanitary Local Department;
HTML=comdrill allows controlling the link of each Municipal area;
HTML_LEGEND=drilleg2 allows control of the link for the legend.
A macro has been used to define the *comdrill* variable inside the dat.indice data set.
For each Commune the HTML HREF instruction is of this kind:
HREF="combody.html#C0xxxxx"
where xxxxx is the Commune Code.
*/

```
   proc gmap data=dat.indice map=map.umbsort;
        by usl;
        id comres;
        block indpct / coutline=green
                       des="Map Usl"
                       html=comdrill
                       html_legend=drilleg2
                       name="USL01"
                       legend=legend1
                       cblkout=same
                       shape=prism;
   run;
   quit;
```

/*
open the body html file for the print output that will be displayed on the first graph legend; *anchor="leg"* is the anchor name, the same as it was used in drilleg variable inside the HREF HTML statement
*/

```
   ods html body="legbody.html"
            anchor="leg"
            path=odsout
            nogtitle;
   goptions notransparency border;
```

/* define title and footnote */

```
   title1
     "Number of Patients Discharged, Residents";
   title2
     "and Hospitalizing Index in the four";
   title3
     "Regional Sanitary Local Departments";
   footnote " ";
```

/*
create the report you can view clicking on the Regional Map legend
*/

```
   proc print data=dat.drillusl noobs;
        var usl discharged resident index;
   run;
   quit;
```

/*
generate the output that is visualised when you click on the block created with proc GMAP. There is a printed output for each BY variable. The correct link is established with the unique name "C054001" (Municipal Code), incremented by one on the base of the BY statement
*/

```
   ods html body="combody.html"
            anchor="C054001"
            path=odsout;
   proc sort data=dat.inddrg out=dat.inddrg;
        by comres descending discharged;
```

/*
dat.inddrg data set contains the index variable for each DRG in each Commune.
*/

```
   title 'Detailed Municipal Data';
   footnote ' ';
   proc print data=dat.inddrg noobs;
        var usl comres drg discharged index;
        by comres;
   run;
   quit;
```

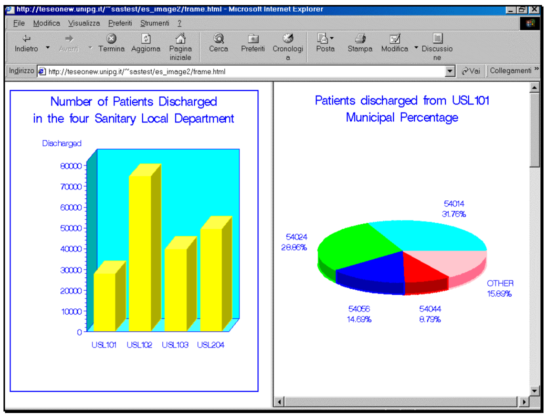
/*
generate the output that is visualized when you click on the USLMAP legend
*/

```
   ods html body="legbody2.html"
            anchor="usl1"
            nogtitle
            path=odsout;
   option nobyline;
   goptions colours=(blue, lime, yellow, red,
                    green, bigy, orange);
   title h=2.5 c=blue
      "Patients Discharged from USL#byval(USL)";
   title2 h=2 c=blue "Municipal Percentage";
   proc gchart data=dat.indice;
        pie3d comres /freq=discharged ctext=blue
                      type=percent coutline=same
                      descending fill=solid
                      noheading name="PIEUSL"
                      other=5 woutline=2;
        by usl;
   run;
   quit;
/*stop sending output to HTML */
   ods html close;
/* reset the default ODS output */
   ods listing;
```

OUTPUT DATA SET OF SAS GRAPH

This method requires a good knowledge of the HTML language**,** but allows drawing the Web page as you prefer. The most important things you must know are how to design a Web page and how to define the links for the graphs with *drill-down* facilities, using the html language. The GANNO, GCHART, GMAP, GPLOT, GREPLAY and GSLIDE procedures have the new option IMAGEMAP=, creating an *imagemap data*set which contains information about the *drill down* zones of the graph. The SAS macro *%imagemap* writes the image map into the HTML file, referencing the *drill-down* graph.

Using a simple DATA step with PUT statement you can write the html file you need, inside a SAS program.


http://teseonew.unipg.it/~sastest/es_image2/frame.html - Microsoft Internet Explorer
File Modifica Visualizza Preferiti Strumenti ?
Indietro Avanti Termina Aggiorna Pagina iniziale Cerca Preferiti Cronologia Posta Stampa Modifica Discussione
Indirizzo http://teseonew.unipg.it/~sastest/es_image2/frame.html
Vai Collegamenti
Number of Patients Discharged
in the four Sanitary Local Department
Discharged
80000
70000
60000
50000
40000
30000
20000
10000
0
USL101 USL102 USL103 USL204
Patients discharged from USL101
Municipal Percentage
54014 31.76%
54024 28.86%
OTHER 15.89%
54056 14.69%
54044 8.79%

Picture 8

Example 3 is an example creating an image map for a chart graph. The frame is separated into two zones (Picture 8), on the left there is the chart graph and on the right the pie graph you choose clicking on each bar. This is a simple way to organize the Web frame: the same result would be obtained automatically using ODS HTML method.

Example 3

/* define location for GIF files */

```
   filename image "/u1/sastest/WWW/es_image2";
```

/*

define three html files:

- *chart.html* referencing vbar3d graph;
- *pie.html* defining links between each pie3d graphic output and the vbar;
- *frame.html* defining the frame window.

*/

```
   filename chart
   "/u1/sastest/WWW/es_image2/chart.html";
   filename pie
   "/u1/sastest/WWW/es_image2/pie.html";
   filename frame
   "/u1/sastest/WWW/es_image2/frame.html";
/* compile the annomac macro */
   %annomac;
   goptions reset=all gunit=pct cback=white
            ftext=swiss htitle=4 htext=2.7
            device=gif noborder
            colours=(blue,lime, yellow, red,
                     green, bigy, orange)
            gsfname=image gsfmode=replace
            xpixels=470 ypixels=550;
```

/*

create the variable containing HTML HREF tag defining the link

*/

```
   data dat.indice;
        length usl2 $8. links $40.;
        set dat.indice;
   if usl=101 then do;
       links='HREF="pie.html#usl1" TARGET="sx"';
       usl2="USL101";
                   end;
   else if usl=102 then do;
       links='HREF="pie.html#usl2" TARGET="sx"';
       usl2="USL102";
                        end;
   else if usl=103 then do;
       links='HREF="pie.html#usl3" TARGET="sx"';
       usl2="USL103";
                        end;
   else if usl=204 then do;
       links='HREF="pie.html#usl4" TARGET="sx"';
       usl2="USL204";
                        end;
   run;
/* define title, footnote and patterns */
   title1 c=blue
      "Number of Patients Discharged";
   title2 c=blue h=4
      "in the four Sanitary Local Departments";
   footnote h=3 "";
   pattern1 v=solid c=yellow ;
   axis1 label=(" ");
   axis2 label=("Discharged");
/* sort data depending on usl2 variable */
   proc sort data=dat.indice;
        by usl2;
```

/*

create chart for drill-down graph; specifying IMAGEMAP= options, creates *map.mapdata* image map data set

*/

```
   proc gchart data=dat.indice
               imagemap=map.mapdata;
               vbar3d usl2 / sumvar=discharged
                  html=links caxis=blu
                  ctext=blue frame cframe=cyan
                  maxis=axis1 raxis=axis2
                  name="UMBRIA";
   run;
/* define title, footnote and patterns */
   title1 h=4 c=blue
        "Patients Discharged from #byval(usl2)";
   title2 h=4 c=blue "Municipal Percentage";
   footnote " ";
   option nobyline;
```

/*

create a graph for each usl; *name=usl1* is automatically incremented in usl11 usl12 usl13.

*/

```
               pie3d comres /freq=discharged
                  type=percent coutline=same
                  ctext=blue fill=solid
                  descending woutline=2 other=5
                  noheading name='usl1';
        by usl2;
   run;
   quit;
```
/* generate html file for *drill-down* graphs */
```
   data _null_;
        file chart;
   put '<HTML>';
   put '<HEAD>';
   put '<TITLE> CHART graph';
   put '</TITLE>';
   put '</HEAD>';
   put '<BODY>';
   put '<BASE TARGET=view_usl>';
   put '<IMG SRC="umbria.gif" '@;
   put ' USEMAP="#UMBRIA">';
```
/*
invoke the *imagemap* macro writing image map to *chart.html* file
*/
```
   %imagemap(map.mapdata, chart);
   data _null_;
        file chart mod;
   put '</BODY>';
   put '</HTML>';
```
/* generate html file to display pie charts */
```
   data _null_;
        file pie;
   put '<HTML>';
   put '<HEAD>';
   put '<TITLE> PIE graph';
   put '</TITLE>';
   put '</HEAD>';
   put '<BODY>';
   put '<A NAME="usl1"></A>@';
   put '<IMG SRC="usl1.gif"><BR><BR>';
   put '<A NAME="usl2"></A>@';
   put '<IMG SRC="usl11.gif"><BR><BR>';
   put '<A NAME="usl3"></A>@';
   put '<IMG SRC="usl12.gif"><BR><BR>';
   put '<A NAME="usl4"></A>';
   put '<IMG SRC="usl13.gif"><BR><BR>';
   put '</BODY>';
   put '</HTML>';
```
/*generate html file to display frames */
```
   data _null_;
         file frame;
   put '<HTML>';
   put '<HEAD>';
   put '<TITLE> Example 3';
   put '</TITLE>';
   put '</HEAD>';
   put '<FRAMESET COLS="50%, *">';
   put '<FRAME SRC = "chart.html" ' @;
   put ' NAME="view_usl">';
   put '<FRAME SRC = "pie.html" NAME="sx">';
   put '</FRAMESET>';
   put '</HTML>';
   run;
```

CONCLUSIONS

SAS/GRAPH software Version 8 provides different methods of publishing graphs on the Web. Each method has its advantages and disadvantages: the easiest is the least flexible, but if you can obtain what you really need it is the best way. If you need to customise your Web page and produce *drill-down* graphs you need to use the IMAGEMAP solution, but a familiarity with HTML language is strictly required. The ODS HTML method combines the best characteristics of the other two methods, because it is quite easy to use and flexible.

Quite remarkable is the *drill-down* facility on the GMAP procedure, a very incisive way of showing data on maps. In fact you can visualise territorial zones in sequence and get every time more detailed information. Moreover you often need to modify the legend in order to reduce the number of colours displayed: the HTML_LEGEND= option solves the problem, allowing, for example, to display a detailed report clicking on the legend.

REFERENCES

SAS OnlineDoc Version 8, SAS Institute

Luca Pasquinucci, *"Introduzione all'Output Delivery System",* Atti del Convegno SUGItalia'99. Roma, 13-15 ottobre 1999

Himesh Patel, Revised by David Caira, *"Using SAS/GRAPH® Software to Create Graphs on the Web",* Proceedings of the Twenty-Fourth Annual SAS User Group International Conference. Miami Beach, Florida April 11-14, 1999

Connie X. Li, James Sun *"Graphically Exploring Multidimensional Data in a Web Browser using new features from SAS/GRAPH®",* Proceedings of the Twenty-Fourth Annual SAS User Group International Conference. Miami Beach, Florida April 11-14, 1999

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Francesca Pierri
Centro d'Ateneo per i Servizi Informatici
Università di Perugia
Via G. Duranti 1/A, S. Lucia Canetola
06125 Perugia, Italy
Work Phone: +39 075 5853794
Fax: +39 075 5853615
Email: frc@unipg.it

TRADEMARKS