

Graphically Exploring Multidimensional Data in a Web Browser -- using new features from SAS/GRAPH

Connie X. Li, Romac International Inc.
James Sun, Constat Systems Inc.

Introduction

The advantage of paperless distribution of reports through internet or intranet is obvious. We would argue that the web technology is one of the best tools in delivering and disseminating information.

In real world, data often reflect a complex relationship among many factors. Those factors are often inter-related. Presenting such a multidimensional data posts challenges in a typical 2D display space. In text format, many have tried to tabulate data with split columns and rows in order to squeezed in more than two dimensions (or factors). In graphic world, a similar approach is applying overlay to place many images as we can. As a result, we often get cluttered images which often bury the pivotal relationship we want to explored.

To overcome the shortcoming of such an approach, we need an interactive method to retrieve information when we need. The drill-down mechanism not only preserves the precise information about relationship among primary factors, but allows user to uncover important patterns in the data for those non-primary factors as well.

Output Delivery System and New SAS/GRAPH

The Output Delivery System (ODS) implements and controls the formatting of all SAS procedure outputs. All procedures produce ODS output object. Within this object, there are two components control output data and output template. The first component moves procedure output into ODS. The second one dictates format and appearance when the output is data rendering. Currently with ODS, SAS output supports HTML and other widely used formats. You can create Web capable graphic file format directly with appropriate ODS statement.

Because the supporting of HTML, the process to create web enabled SAS output is much more convenient and

transparent. It allows creation of HTML documents in either batch or interactive mode without thorough knowledge of HTML

Before version 7, SAS/GRAPH can be used to generate basic web capable images with some experimental device drivers. But those preliminarily graphic drivers have no capability in embedding hyperlink into the graphic outputs, which is essential to bring 'hot spot' linking mechanism to live within a browser environment.

As one important enhancement for new SAS/GRAPH, some of procedures allow user to create position-sensitive image. For example, a HTML option statement in PROC GCHART can generate position-sensitive hyperlinks for different graphic areas.

By using `HTML=href` option statement, the graphic output has taken the input of `href` (from the input dataset) created hyperlinks behind the certain area of image, (i.e., a bar area from a bar chart), to a designate HTML document. The magic is accomplished with HTML tags `<MAP>` and `<AREA HREF>`

Multiple Frames

The approach described here is using multiple frames to display different graphics. One of the frames will present a main graphic image which has hyperlinks built for other dimensions. The rest of frames are ready for other dimensional graphics. When user click the portion of main image associated with a dimension, a set of graphics for the dimension will be displayed around the main frame.

Here is a sample frame partition (**Figure1**). It can be used to present a main graphic and other five graphics. The number of surrounding small frames can be changed to accommodate practical situation. The HTML code for such partition is listed in Appendix A.

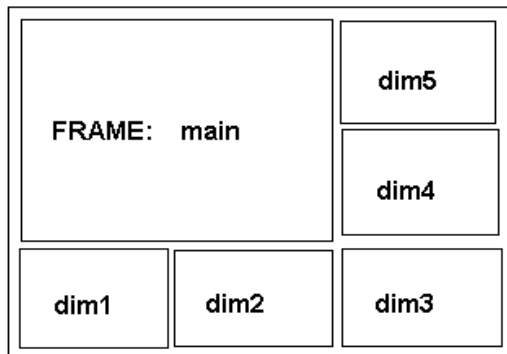


Figure 1. Frame Partition of Browser Window

In order to control the behavior of all those surrounding frames from the main graphic, javascript is used. This javascript function will alter the default hyperlink behavior, which will only load file in the same frame. Now we can control surrounding frames by clicking certain area of main graphic.

This piece of javascript is placed inside the HTML code generated for main graphic 'sugi24_m.htm'. So the ODS statement before main graphic is appended to the Javascript section.

Example

As an example we use a fictitious stock holding and its performance data. It consists of seven variables. STOCK, represents the stock ticket symbol; VOLUMN, is the current number of shares. QUART, is a time point variable; PRICE, represents stock price; EPS, BETA, PE are the other performance variables. Each stock has its performance data collect at eight time points. The table below shows the part for only one stock 'MRC'.

STOCK	QUART	VOLUMN	PRICE	EPS	BETA	PE
MRC	98Q4	190000	72.5	0.98	1.0	33.4
MRC	98Q3	205000	75.1	0.90	1.1	31.2
MRC	98Q2	210000	83.5	0.93	1.0	29.7
MRC	98Q1	210000	79.3	0.96	1.2	38.2
MRC	97Q4	255000	98.2	1.05	1.1	39.2
MRC	97Q3	240000	88.5	1.09	1.1	33.6

As discussed above, we need to create *href* for the HTML = options, in order to generate hyperlinks for different area of the pie chart. The

following SAS code shown below illustrates the way to construct data sensitive link information. ('num' is relate to the 'stock'). Depend on the data order, each stock has its unique hyperlink. When we later construct pie chart, those hyperlinks are inserted behind the slices representing different stocks.

```
data stocknow;
  set holding end=eof;
  by stock quart;
  invest=price*volumn;
  if last.stock then do;
    num+1;
    output;
  end;
  length file $80 href $80;
  file = "JavaScript:twoinone(" ||
    put(num, 1.0) || ")";
  href = 'href="' || trim(file) || ''';
run;
```

Code 1. Data Preparation for Main Pie Chart

Main Graphic

The main graphic is the one showing current stock holding portfolio. It is the primary focal point for user in this case. A 3D pie chart is used to present the proportion of capital invested among various stocks.

The first chunk of SAS code is for embedding javascript function into the main graphic's HTML file. It first retrieves the javascript then output into the main HTML file.

```
ods html close;

filename sugi24 'c:\sugi24_m.htm';
filename javascript 'c:\javascript.txt';

data _null_;
  infile javascript;
  file sugi24;
run;

ods html close;

filename sugi24 'c:\sugi24_m.htm' mod;
ods html dev=gif body=sugi24 (no_top_matter)
```

```

title .

goptions target=gif device=html
        gunit=pct ftext=swiss
        htitle=6 htext=3
        hsize=6 in vsize=4.5 in;

legend1 label=none position=(top center
outside) mode=share;

pattern1 v=s c=blue;
pattern2 v=s c=cyan;
pattern3 v=s c=green;
pattern4 v=s c=magenta;
pattern5 v=s c=red;

proc gchart data=stocknow;
  title1 h=5 pct 'Investment Portfolio';
  pie3d invest / noheading
  html=href
  name="stock" ;

run;
quit;

ods html close;

```

Code 2. SAS Code for Main Graphic

After placing javascript code, we are using PROC GCHART to produce the main graphic. The body=sugi24 (no_matter_top) points to a filename reference 'sugi24'. And the option next to the file reference indicates no standard HTML file heading is needed. The 3D pie chart option is a new feature. It presents a pretty pie chart with 3D appearance. Together with other 3D bar options, PROC GCHART emerges as a great contender in generating business presenting quality graphics tool.

Supporting Graphics

The supporting graphics are the charts presenting stock performance variables vs. time points. They are generated separately with PROC GCHART procedures. At first, performance data for one stock will display in the surrounding frames. User can then click the pie chart to bring up the performance charts for one of the stock respectively. For appearance purpose, all supporting graphics chose different bar shapes.

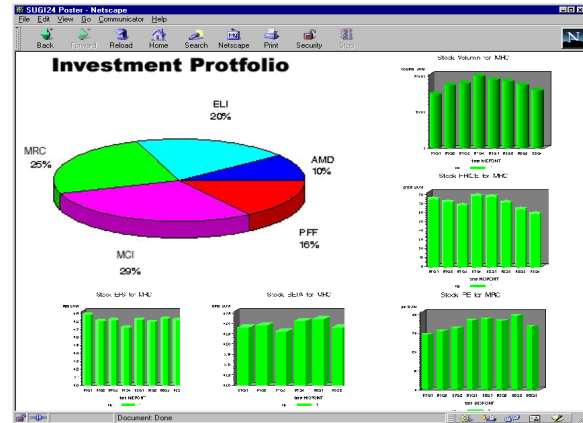


Figure 2. The Appearance of Browser Window

The figure above is the appearance of the browser window.

Conclusion

With the development of version 7 SAS, especially Output Delivery System, web browsers become the choice in presenting business information stored in a SAS based system. This paper presents a usage of new features in SAS/GRAPH in order to presenting multidimensional data in a web browser environment. Using new features, a set of graphics are generated with hyperlinks embedded reflect the relationship among individual graphic. It makes the process of exploring multidimensional data much more intuitive.

Reference

SAS Institute Staff, SAS version 7 beta OnlineDoc, Cary, NC, 1998

SAS Institute Staff, What's New for Version 7 SAS Software Products, <http://www.sas.com/service/doc/v7/whatsnew/>, Cary, NC, Dec.1998

Contact:

Connie Li
83 Regal Drive
Monmouth Junction, NJ 08852
Email: liconnie@mars.superlink.net

Appendix A HTML Code for Frames Partition

```

<html>
<head><title> SUGI24 </title></head>
<frameset rows='400,*' FRAMEBORDER =0 BORDER = 0 FRAMESPACING = 0>
  <frameset cols='500,*' FRAMEBORDER =0 BORDER = 0 FRAMESPACING = 0>
    <FRAME SRC = 'd.htm' NAME = 'main' border=0 NORESIZE MSCROLLING =
NO MARGINWIDTH = '0' MARGINHEIGHT = '0' >
      <frameset rows='200,200' FRAMEBORDER =0 BORDER = 0 FRAMESPACING =
0>
        <FRAME SRC = 'mrc_beta.htm' NAME = 'dim5' border=0 NORESIZE
SCROLLING = NO MARGINWIDTH = '0' MARGINHEIGHT = '0' >
          <frame src = 'mrc_pe.htm' NAME = 'dim4' border=0 NORESIZE
SCROLLING = NO MARGINWIDTH = '0' MARGINHEIGHT = '0' >
            </frameset>
          </frameset>
        <frameset cols='265,265,*' FRAMEBORDER =0 BORDER = 0 FRAMESPACING = 0>
          <FRAME SRC = 'mrc_price.htm' NAME = 'dim1' border=0 NORESIZE
SCROLLING = NO MARGINWIDTH = '0' MARGINHEIGHT = '0' >
            <FRAME SRC = 'mrc_volumn.htm' NAME = 'dim2' border=0 NORESIZE
SCROLLING = NO MARGINWIDTH = '0' MARGINHEIGHT = '0' >
              <frame src = 'mrc_eps.htm' NAME = 'dim3' border=0 NORESIZE
SCROLLING = NO MARGINWIDTH = '0' MARGINHEIGHT = '0' >
                </frameset>
              </frameset>
            </body>
          </body>
        </html>

```

Appendix B Javascript Code (javascript.txt) for Controlling Surrounding Frames

```

<SCRIPT>
<!--
function twoinone(nr){
  if (nr==1){
    parent.dim1.location.href="mrc_price.htm"
    parent.dim2.location.href="mrc_volumn.htm"
    parent.dim3.location.href="mrc_eps.htm"
    parent.dim4.location.href="mrc_beta.htm"
    parent.dim5.location.href="mrc_pe.htm"
  }
  if (nr==2){
    ..... (omit)
  }
  if (nr==3){
    ..... (omit)
  }
  if (nr==4){
    ..... (omit)
  }
  if (nr==5){
    ..... (omit)
  }
}
//-->
</SCRIPT>

```