

Dynamic Drillable Map on Internet using Output Delivery System and SAS/Graph®

Hsiwei Yu, Digital Intelligence Systems, Centreville, Virginia

Chapman Gleason, U.S. Environmental Protection Agency, Washington, DC

ABSTRACT

Combining new features from the Output Delivery System (ODS) and SAS/Graph in Version 8, it is possible to create static drillable maps, not only hyper-linked for each region but also for annotated geographical points. We further developed techniques to send the resulting image and HTML file directly to the Internet, achieving dynamic drillable maps.

INTRODUCTION

We have always wanted to present an U.S. map, showing a metric, say income, of all fifty states. The user clicks on a state then more information about that state is shown, perhaps a state map of counties showing each county's measurement plus other relevant textual reports. Larry Hoyle had developed a Java program for making drillable maps (Ref 1). By using products from Environmental Science Research Institute®, Myles Powers developed drillable map with zoom-in, zoom-out, and other features (Ref 2) for display over the Internet. Before Version 7, SAS provided a macro program for translating a map dataset into HTML AREA tag statement. However this macro program must be run in two phases, as such, it is static and not dynamic (Ref 3).

ODS readily supports drillable action in Proc Gmap, Gchart, and Gplot. There are sample programs on SAS Technical Support web site (Ref 4) and SUGI papers (Ref 5) illustrating how to make a bar or pie chart, or legend in a graph drillable. These samples showed how to pre-generate static HTML files. Here we will explore ways to dynamically generate drillable maps for the Internet.

PROBLEM STATEMENT

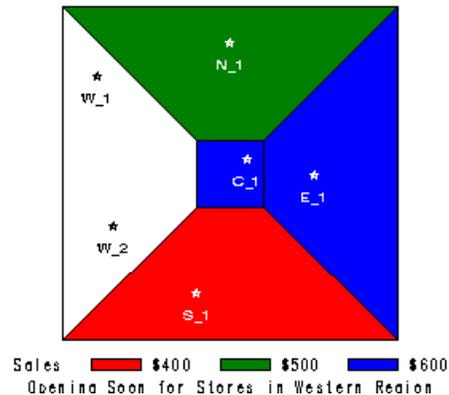
Given sales figures for some retail stores and their geographic coordinates, display on the Internet a choropleth map showing sales volume statistics and annotate the map with store locations. Allow dynamic drill-down into each store and each sales region. Consider the sample data in the table below:

Sales Data for Each Store

Obs	Region	Store Name	Sales	Note
1	North	N_1	\$500	
2	South	S_1	\$400	
3	Central	C_1	\$600	
4	East	E_1	\$700	
5	West	W_1	.	Opening Soon
6	West	W_2	.	Opening Soon

Note the first four stores are already in operation, and the last two stores are planned to open soon. The longitude and latitude coordinates for these stores are not shown to conserve space. It is easy to derive regional sales statistics, e.g. sum, average, etc, from this store information. For illustration purpose, below is a map showing sales region and store locations:

Store Location and Sales Volume



HTML CLIENT-SIDE IMAGE MAP AND OUTPUT DELIVERY SYSTEM

The MAP and AREA statements in HTML supports client-side image map, e.g.

```
<html>
<!-- This HTML file is in
http://.. path1/HTML FileName.htm -->
<head><title> ..</title></head>
<body> ..
..

<map name="name">
<area shape="poly" href=".." coords=".. , ..">
<area shape="rect" href=".." coords=".. , ..">
..
</map>..
</body> </html>
```

ODS can create the HTML file containing the appropriate MAP and AREA statements based on SAS/Graph. Note two files are needed in this approach, the HTML file and the graphical image file (referenced by the HTML file).

STEPS FOR WEB ENABLED SOLUTION

This problem can be solved using ODS, SAS/Graph, and SAS/IntrNet™ in these steps,

- 1) Use ODS and SAS/Graph to create a HTML file and map image file.
- 2) With the help of specific ODS features make sure these files on the executing hosts are accessible from the web server machine.
- 3) Send the generated HTML file back to the surfing visitor.
- 4) Remove intermediate files on the machine where SAS/IntrNet executing.

A significant restriction is that the web server and host running SAS/IntrNet share a same file system then the generated HTML and graphic files can easily sent back to user, achieving dynamic results. It's still possible to overcome the handicap of web server and host executing SAS/IntrNet having different file systems. However it is outside scope of this paper. As of this writing, SAS and SAS/IntrNet for Version 8 is not yet available to this author.

However, we used a Perl program substituting for the real SAS/IntrNet (Ref 6).

STEP ONE: DRILL ON ANNOTATED POINTS AND REGIONS

Each annotated point on a map can represent a drill down action. It is a requirement that the action be specified in the annotate dataset's variable named 'html'. Each region in a map can represent a drill down action. This action is specified as a variable, say 'RegionDrill', in the data dataset. Then in the choropleth map statement, the html option is given that variable, say 'RegionDrill'. However, if we want *both* annotated points and regions in a map drillable, the annotate dataset dictates that the data dataset must name its variable as 'html' for drill on regions.

With the aforementioned restrictions in mind, we can proceed to construct the data and annotate datasets. The outline for Proc Gmap is something like this:

```
Proc gmap data= RegionalSalesStatistics
  Map= RegionalMap
  Anno= AnnotateStoreLocation;

  Id Region;
  Choro Sales / html= html ..;
```

DRILL ACTION ON ANNOTATED POINTS

Construct the annotate dataset as usual, except that a new variable named 'html' is required for the drill action.

```
data AnnotateStoreLocation;
  set StoreData( keep= x y StoreName);
  /* X and Y are the latitude and longitude
  coordinates */
  length html $ 300; /* As long as needed */
  style= 'special'; /* Font */
  text= 'I'; /* 'I' becomes a star, '*' */
  color= 'black';
  position= '5'; /* Right in the middle */

  ** Drill down action on this annotated point;
  ** This variable must be named 'html';
  html= 'href="/cgi-bin/broker?StoreName='
    || StoreName || ' .. ..';
  output;
  .. ..
Run;
```

DRILL ACTION ON REGIONS

For making regions in a map drillable, construct a variable in the data dataset representing the drill action, then in the Proc Gmap assign it the HTML= option. For making legends in a map drillable, construct a variable representing the drill action, then in the Proc Gmap assign it the HTML_LEGEND= option. However, programmer doesn't have explicit control over how the legend will be constructed by SAS/Graph, making this legend option less useful.

```
data RegionalSalesStatistics;
  /* Required name when in conjunction with
  annotated points drilling */
  length html $ 300;

  /* Drill action when clicking on a region. */
  html= 'href="/cgi-bin/broker?'
    || 'Region=' || Region
    || ' .. .."';

  /* If you want null action when clicking on a
  region,
  html= 'href="JavaScript: ; "' ;
  */
  output;
```

```
run;
```

PROC MAP

With the data and annotate datasets properly constructed, now show it in a choropleth map.

```
Proc gmap data= RegionalSalesStatistics
  Map= RegionalMap
  Anno= AnnotateStoreLocation;

  Id Region;
  Choro Sales / html= data one;
Run; quit;
```

STEP TWO: SAVE FILES IN THE RIGHT LOCATIONS

Output Delivery System actually constructs HTML file with drill down information using MAP and AREA tags,

```
<html>
<!-- This HTML file is in
http://.. path1/HTML\_FileName.htm -->
<head><title> ..</title></head>
<body> ..
..

<map name="name">
<area shap=.. href=.. coords= ..>
..
</map>
..
</body> </html>
```

From installation's webmaster, we can obtain cross reference of directories between web server machine and host executing SAS/IntrNet. For example,

Web Server's Directory	SAS/IntrNet Host's Directory
http://www.xyz.com	/inetpub/www
http://www.xyz.com/salesDept	/inetpub/www/sales

We can use ODS statements to capture both the image file and the corresponding HTML file.

```
** HTML file having the drill-down info.;
%let multi= 8; %let lng_k= 1024;
%let LRECL= %eval( &multi * &lng_k );
** Ensure maximum record length for complex
regions.;
Filename htmlFile 'path_a/HTML_FileName.htm'
  Lrecl= &LRECL;

** Capture the HTML file;
Ods html file= htmlFile;

** A directory where graphic image file will
be saved.;
Filename imageFil 'path_b';

** Tell SAS/Graph to create the GIF file.;
Goptions gsfname= imageFil gsfmode= replace
  Device= gif;

** Use base= to give the image file's full
path name. Note the ending slash, '/', in the
base option is important!;
Ods html Base= 'http://.. path2/';

Proc gmap data= RegionalSalesStatistics
  Map= RegionalMap
  Anno= AnnotateStoreLocation;
```

```

Id Region;
Choro Sales / html= data one;
Run; quit;

```

Also note if a geographic area is very complex, SAS/Graph may run into problems. Try not to generate very complicated graph, e.g.

```

Proc gmap map= maps.counties
    ( where= ( density < 3 ) )
.. ;
id county;
choro county;

```

Chapman Gleason
U.S. Environmental Protection Agency
Washington, D.C.
Work Phone: 202-260-9006
Email: gleason.chapman@epa.gov

STEP THREE: SEND HTML FILE TO SURFING VISITOR

Once the HTML file is properly constructed, we can use a data step to send it out to Internet.

```

Data _null_;
Infile htmlFile;
/* Fileref _webout is reserved for sending
back to the client in SAS/IntrNet */
File _webout;
Input ;
Put _infile_;
Run;

```

STEP FOUR: DELETE WORK FILES

We can devise some mechanism to delete the generated HTML and graphic image files, such as schedule a Unix cron job or Window/NT at command.

CONCLUSION

The Output Delivery System allows SAS programmers easily produce drillable maps on Internet. This approach offers immediate results for people not versed in Java or other software packages

REFERENCES

1. "Choosing a Method for Connecting Java to the SAS System across the Internet – CGI, JDBC, Socket, or Jconnect?", Larry Holye, University of Kansas, NESUG 1997, Baltimore, MD
<http://www.ukans.edu/cwis.units/IPPBR/ksdata/ksah/javamap.htm>
2. Myles Powers of Logicon Corporation, presentation in Maryland SAS Users Day, 1999, Baltimore, MD
3. ISMAP.SAS program by Michael Burns of SAS Institute, 1995, previously available from SAS Institute web site.
4. Resources from SAS Institute
http://www.sas.com/service/techsup/sample/sample_graph.html
http://www.sas.com/service/techsup/sample/sample_graphv7.html
http://www.sas.com/software/components/graph_dir/demos.html
5. "Using SAS/Graph Software to Create Graphs on the Web", Himesh Patel, David Caira of SAS Institute, SUGI 24, 1999
6. "Using Sockets in SAS Software for Internet Publishing", Hsiwei Yu and Chapman Gleason, SUGI 24, 1999

CONTACT INFORMATION

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

Hsiwei Yu (Michael)
Work Phone: 202-260-5312
Email: vhyu@netkonnnect.net (H)
yu.michael@epa.gov (W)