Building Drill-down SAS® Applications

Kirk Paul Lafler, Software Intelligence Corporation
Charles Edwin Shipp, Shipp Consulting

Introduction

In a world where business decisions must be made quickly, an application must be able to deliver information in a blink of an eye. Trends, facts, and figures must be translated into meaningful information so its meaning can get inside the heads of people. Decision makers are not satisfied with looking at rows and rows of numbers in an attempt to unmask the hidden meaning of data. Data must be provided visually as in a chart or graph so the picture itself tells the story without a word of explanation.

Data visualization techniques and technologies bring new levels of understanding to your data. The SAS® Output Delivery System *(ODS)* is a marvelous tool for formatting output generated by SAS procedures and DATA steps. ODS enables effective application design strategies to be implemented while providing flexibility and ease of use. Combined with products such as SAS/GRAPH software, new levels of interactivity bring your data to life with drill-down approaches. End users start out with summary information and can dive into the details at the click of their mouse button using a graphical user interface. This article describes the process of building a simple drill-down application with the SAS System. All you need is base-SAS and SAS/GRAPH software.

Drill-down Applications

As a general rule, the best type of user interface design for transaction-based applications is a drill-down user interface opposed to a character-based one. It's referred to as drill-down because a user drills down through the data, layer by layer, until the desired information is found.

Two significant design objectives exist for drill-down user interfaces. The first objective is to permit users to get the information they need quickly in order to make good business decisions. The second objective is to display only that data that is necessary to the user at a time.

The key to building successful drill-down applications requires systems analysts and system designers to understand what users are trying to achieve with the data. These individuals must recognize the tasks users engage in while trying to access the desired information. These tasks are then translated into a series of selection criteria that users should be able to select from.

Drill-down with Layered Charts

As the amount of data available on the Internet grows exponentially, systems analysts and designers are faced with huge challenges. First, how should meaningful information be extracted from data? Second, what is the best way to keep the information current in an ever-changing world of information?

A popular method of visually presenting data while bringing out significant trends and outliers is accomplished with charts. Data is frequently displayed as pie charts, vertical and horizontal bar charts, geographical maps, as well as a variety of other visual displays. Besides offering visual trends of data, charts provide convenience and offer a standard way of performing data analysis through drill-down navigation.

Drill-down approaches are extremely flexible, allowing any chart to act as a graphical front-end interface to additional graphical or textual information. Data is frequently displayed by layering one chart on another providing an effective point-and-click user interface. At the top layer data is presented in a highly summarized fashion. As the user clicks on a designated area of the chart, data is displayed in significantly more detail. The user navigates down the various layers of data with one or more clicks of the mouse or a combination of cursor and keyboard commands to see more and more details. It's simple and, above all else, elegant.

Creating PDF Output

To share output electronically, Adobe created a proprietary format called PDF. The objective of PDF is to enable the printing of output exactly as it is seen. The significance of PDF output is that it is a great format for Web deployment since it is completely independent of any printer destination. To create PDF output, the ODS PDF option is specified as follows.

Code

```
ODS Listing Close;
ODS PDF FILE='ods-univariate.pdf';
  proc univariate data=libref.movies;
    title1  'Creating PDF Output with ODS';
  run;
ODS HTML Close;
ODS Listing;
```

Links and References in ODS

The ODS HTML statement controls how links and references are constructed between one or more HTML destination files. The basic syntax of the HTML destination follows:

> **ODS HTML ODS-action;**
>
> < or >
>
> **ODS HTML HTML-file-specification < options >;**

When an ODS-action is specified, one or more output objects are selected or excluded, or the HTML destination is closed. The available ODS-actions are: 1) CLOSE, 2) EXCLUDE, 3) SELECT, and 4) SHOW. When an HTML-file-specification is specified, ODS routes one or more pieces of output to a designated file or files. The available options include:

```
ANCHOR='anchor-name'
BASE='string'
GFOOTNOTE <or> NOGFOOTNOTE
GPATH=file-specification
GTITLE <or> NOGTITLE
HEADTEXT='HTML-document-head'
METATEXT= 'HTML-document-head'
NEWFILE=NONE <or> OUPUT <or> PAGE <or>
PROC
PATH=file-specification
RECORD_SEPARATOR='string' <or> NONE
STYLE='style-definition'
TRANTAB='translation-table'
```

Combining Output Results

With the streaming capabilities of HTML output, results can be combined so they appear on the same screen. Rather than having output controlled by one or more page breaks, HTML automatically displays output without page boundaries. The following example code illustrates combined output from the PRINT and MEANS procedures.

Code

```
ods html path='c:\sas app'
       body='ods-body-combined.html'
    contents='ods-contents-combined.html'
       page='ods-page-combined.html'
       frame='ods-frame-combined.html';
  proc print data=odslib.movies  noobs  n;
    title1 'Movie Classics Listing';
    where rating in ('G', 'PG');  run;
  proc means data=odslib.movies;
    title1 'Summary of Movie Clasics';
    class rating;
  run;
ods html close;
```

Output from Combining Output

Changing Output Labels

Specifying the PROCLABEL option lets you change the "default" label displayed on procedure output. To change the label produced by the UNIVARIATE procedure from "The Univariate Procedure" to "Movie Classics Statistics", the following code is specified.

Code

```
ods html path='c:\sas app'
       body='ods-body-label.html'
       page='ods-page-label.html'
    contents='ods-contents-label.html'
       frame='ods-frame-label.html';
ods proclabel 'Movie Classics Statistics';
proc univariate data=odslib.movies;
  title1 'Creating HTML Output with ODS';
  title2
    'HTML FRAME File with Changed Labels';
run;
ods html close;
```
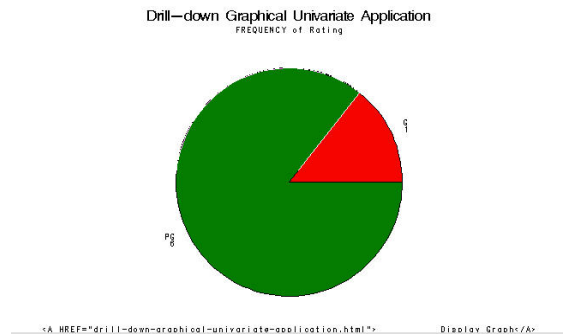
Building a Drill-down Application

To simplify the process of building a graphical drill-down application in the SAS System, five easy steps are presented.

1. Create a data set containing the location of the HTML link variable.
2. Create HTML path with BODY and optional files.
3. Create graph using HTML= option and link variable.
4. Create detail list drill-down pages.
5. Use Web browser to navigate through resulting application.

Example Drill-down Application

In the following drill-down application, a pie chart is used to display summary information on G-rated and PG-rated movies. A horizontal or vertical bar chart could also be used as the graphical user interface. To display detailed information on a movie category, a user would only need to click on the desired piece of the pie chart that they had interest in, as shown in the illustration below. Control would then be passed, via hyperlinks, to the underlying detail output created in step #4 above.

Example Code

The SAS coding steps used to in build a drill-down application under the Windows® platform is presented below.

```
*STEP 1 - Define data set and link variable;
data movie_ratings_g_pg;
  set odslib.movies;
  length urllink $30.;
  if upcase(rating) = "G" then
     urllink="href=g-ratings-body.html";
  else if upcase(rating) = "PG" then
     urllink="href=pg-ratings-body.html";
run;

*STEP 2 - Create HTML path and BODY file;
ods html path='c:\sas app'
        body='drill-down-graphical-
application.html';

*STEP 3 - Create graph using HTML= option;
goptions device=gif hsize=7in vsize=4in;
proc gchart data=movie_ratings_g_pg;
  title1 "Drill-down Graphical Application";
  pie rating / html=urllink;
  where upcase(rating) in ('G', 'PG');
run;
quit;

*STEP 4 - Create G-rated drill-down list;
ods HTML path='c:\sas app'
        body='g-ratings-body.html';
proc univariate data = movie_ratings_g_pg ;
 title1 'Creating a Drill-down Application';
 footnote1  '<A  HREF="drill-down-graphical-
application.html">
           Display Graph</A>';
  where upcase(rating)="G";
run ;
ods HTML close;

*STEP 5 - Create PG-rated drill-down list;
ods HTML path='c:\sas app'
        body='pg-ratings-body.html';
proc univariate data = movie_ratings_g_pg ;
```

```
 title1 'Creating a Drill-down Application';
 footnote1  '<A  HREF="drill-down-graphical-
application.html">
           Display Graph</A>';
  where upcase(rating)="PG";
run ;
ods HTML close;
ods Listing;
```

Testing Drill-down Applications

Before deploying drill-down applications into production, users should thoroughly test all code, including HTML links and returns to make sure they are problem-free. Many viewers will not return to an application that contain errors or does not work according to design. A word of caution*: Not all Web browser software handles web pages the same way.*

Before deploying a drill-down application to a server, the Web, Intranet, or Extranet for others to use, it is important to visually inspect and test its overall behavior and operation. One way to assist in this process is to develop a checklist consisting of important attributes related to operation, process control, errors or inconsistencies in results, and appearance. The following recommendations provide a few items to consider before deploying a drill-down application.

1. Check spelling – check the spelling on each of your application's screens or pages before making them available to users.
2. If possible, take the time to review the application to identify errors in the code, output, and processes or use a validation service.
3. Test the Web pages to see how easy they are to access and browse through the information. You should verify that each Web page has a consistent design and layout.
4. Turn off images to test how Web pages will look and what information is displayed when viewers use Web browsers that cannot display in pages or when they turn off images.
5. Verify links to make sure each link takes you to the intended destination and that each link contains information of interest to your viewers.
6. Enlist a test audience to check out your Web pages and to solicit their feedback. This feedback is very important since it will enable you to improve the way your Web pages look and operate. It is also important that you compare your test audience's feedback with your own objectives to determine which areas require more work.
7. Test your Web pages with different Web browser software to evaluate how they will look. The two most popular Web browsers are Microsoft Internet Explorer and Netscape Navigator.
8. Test Web pages on different computers because they can look and behave differently when the content consists of animation.

9. Determine the speed of transferring Web page content. If the content is too text-rich or image-rich, the excessive transfer speeds may cause viewers to tune-out rather than tune-in.
10. View your Web pages at different resolutions to determine the amount of information that can be displayed at one time.

Conclusion

With the growing popularity of the Internet, the Output Delivery System (ODS) is extremely useful in deploying selected pieces of output on the Web, Intranet, or Extranet. ODS makes the development of drill-down SAS applications a snap. Users will appreciate the ability to see a visual representation of the data in a custom graphical user interface.

Data visualization techniques and technologies bring new levels of understanding to your data. The Output Delivery System is a marvelous tool for formatting output generated by SAS procedures and DATA steps. ODS enables effective application design strategies to be implemented while providing flexibility and ease of use. Combined with products such as SAS/GRAPH software, new levels of interactivity bring your data to life with drill-down approaches.

Acknowledgments

We would like to thank the WUSS 2002 leadership for accepting this paper, as well as for their hard work and support of a great Conference.

References

Heffner, William F. (1998), "ODS: The DATA Step Knows," Proceedings of the 23rd Annual SAS Users Group International Conference, Cary, NC: SAS Institute Inc.

Lafler, Kirk Paul (2002), "Output Delivery Goes Web," Proceedings of the 27th Annual SAS Users Group International Conference, Software Intelligence Corporation, Spring Valley, CA, USA.

Lafler, Kirk Paul (2000), The SAS® Output Delivery System (ODS) Answer Guide, Revised and Updated, Software Intelligence Corporation.

Lafler, Kirk Paul (1999), "Delivering Results with the Output Delivery System," Proceedings of the 24th Annual SAS Users Group International Conference.

Olinger, Christopher R. (1998), "ODS for Data Analysis: Output As-You-Like-It in Version 7," Proceedings of the 23rd Annual SAS Users Group International Conference, Cary, NC: SAS Institute Inc.

Patel, Himesh (1998), "Using SAS/GRAPH® Software to Create Graphs on the Web," Proceedings of the 23rd Annual SAS Users Group International Conference, Cary, NC: SAS Institute Inc.

SAS Institute Inc. (1999), The Complete Guide to the SAS® Output Delivery System, Version 7-1, Cary, NC, USA.

Wehr, Paul (1998), "Building Clinical Information Spaces on the World Wide Web," Proceedings of the 23rd Annual SAS Users Group International Conference, Ann Arbor, MI: STATPROBE, Inc.

Trademark Citations

About the Authors

Kirk Paul Lafler is consultant and founder of Software Intelligence Corporation with 25 years of SAS programming experience. As a SAS Institute Alliance Member (1996–2002) and SAS Certified Professional, Kirk provides IT consulting services and training to SAS users around the world. As the author of three books including Power SAS by Apress and more than one hundred articles, Kirk is a frequent speaker at International, Regional, and Local SAS User Groups. He also writes Kirk's Korner, a popular SAS Tips column that appears regularly in the BASAS, HASUG, SANDS, and SESUG Newsletters. He can be reached at KirkLafler@cs.com.

Charles Edwin Shipp is a Senior Programmer and Consultant with over 25 years experience working with the SAS System. He has authored numerous articles and co-authored the popular Books by Users (BBU) book, Quick Results with SAS/GRAPH Software. His expertise includes application design and development, training, and programming using base-SAS, JMP, SQL, ODS, SAS/FSP, SAS/AF, SCL, FRAME, and SAS/GRAPH software. He has also been involved with web design, content creation, and development.

The authors welcome comments and can be reached at:

Software Intelligence Corporation
P.O. Box 1390
Spring Valley, CA 91979-1390
E-mail: KirkLafler@cs.com