

# Output Delivery System Tips and Techniques

Kirk Paul Lafler, Software Intelligence Corporation

## Abstract

Are you looking for ways to improve the way your output appears? The SAS<sup>®</sup> Output Delivery System (**ODS**) can help turn tired-looking output into great looking information with a purpose. ODS introduces exciting new features for your output. Gone are the days when the only available formatting choice is basic output listings consisting of monospace fonts printed on a white background. Using built-in format engines, ODS addresses the inherent weaknesses found in traditional SAS output by enabling “quality” output to be produced without having to import it into word processors. In Version 9, many new output formatting features and options are introduced for SAS users to take advantage of. Users have a powerful and easy way to create and access formatted procedure and DATA step output.

## Introduction

Technology is moving at an astounding pace. With many new features and enhancements introduced in Version 8, users can finally sound the trumpets for what could be the most exciting time for output delivery since the introduction of the color monitor and printer. Never again will SAS users be confined to boring monospace output. Instead, output delivery has entered a new age taking full advantage of font characteristics, color, a variety of output layouts, and numerous other features. This paper and presentation will treat users with the many new and exciting features found in ODS. Sprinkled throughout will be numerous tips, tricks, and techniques that will, hopefully, make the learning process a little easier.

Attendees will learn numerous tips, tricks, and techniques in handling output including the advantages / disadvantages with traditional SAS output; advantages of ODS; how ODS handles raw data; how ODS combines raw data with table definitions; how to open, close, and manage output destinations; how the Results window stores links to ODS output; how selection and exclusion lists are used; how to create output data sets, RTF files, HTML output, and postscript files; and how ODS is integrated with the DATA step.

## Output Delivery Basics

The SAS<sup>®</sup> Output Delivery System (**ODS**) provides many ways to format output. It controls the way output is accessed and formatted. Although ODS continues to support the creation of traditional SAS listing or monospace output (i.e., Listing), it provides many new features and greater flexibility when working with output.

An early version of ODS appeared in Version 6.12 as a way to address the inherent weaknesses found in traditional SAS output. It enables “quality” looking output to be produced without having to import it into word processors such as MS-Word. In Version 8 and 9, many new output formatting features and options are introduced for SAS users to take advantage of. Users have a powerful and easy way to create and access formatted procedure and DATA step output.

### **Tip #1 – ODS and “Batch” Use**

Many of the ODS features found in the interactive side of the SAS Display Manager System (DMS) can also be used in batch processing. ODS has been designed to make exciting new formatting options available to users. In a windowing environment, ODS can send output to the following destinations: the output window (DMS), the listing file, HTML, SAS dataset, rich text format (RTF), postscript file, external output file (non-SAS file), or output device. The only exception for batch processing is having output sent to the output window.

### **Tip #2 – What if I’m Still Not Using the latest Version of SAS Software**

First introduced in Version 6.12, ODS offered users the capability to format output to destinations other than traditional line printers. Version 6.12 introduced the ability to deploy output to the web, the creation of SAS datasets and rich text format (RTF) files, and DATA step interaction. ODS was designed to address the inherent weaknesses found in traditional SAS output. It enables the creation of “quality” looking output without having to import it into word processors such as MS-Word. New output enhancements were introduced in Version 8 and then in Version 9, including the ability to create postscript files and output customizations. To take full advantage of the power offered in ODS, it is recommended that users upgrade to the latest Version as early as possible to take advantage of these features.

### **Tip #3 – ODS and System Resources**

A very important efficiency consideration is to remember that ODS currently supports the following destinations: 1) Listing, 2) HTML, 3) rich text format (RTF), 4) postscript, and 5) Output. (Note: It also provides support in the DATA step.) Each ODS destination can be open or closed at the same time. For each open destination, ODS sends output

object(s) to it. System resources are used when a destination is open. As a result, make sure any and all unwanted open destinations are closed to conserve on resources.

#### **Tip #4 – Closing Destinations before and after use**

The Listing destination is open by default at SAS invocation, while the other destinations are closed. If nothing is done to suppress output to the Listing destination, your SAS programs automatically produce Listing output, just as they always have in the SAS System. If you needed to suppress printed output from being sent to the Listing destination (or DMS Output window) before the execution of a procedure step, the following ODS statement would be issued:

```
ODS Listing Close;  
Proc univariate data=libref.movies;  
Run;  
ODS Listing;
```

By closing the Listing destination before the procedure code, the SAS System is actually suppressing output to that destination until it is reopened. The preceding example shows that at the end of the procedure step, the Listing destination is reopened by specifying ODS Listing; so output from subsequent steps can be sent to the Listing destination.

#### **Tip #5 – Deleting Output from the Results Window**

The Results window identifies procedure output that has been produced, providing users with an improved way to manage their output. It is customarily a good thing to remove unwanted output displayed in this window to conserve on system resources. The Results window is opened by specifying the command **ODSRESULTS** on the DMS command line or by selecting **View Results** from the pull-down menu.

To delete procedure output, use the following steps:

1. Select the procedure folder you want to remove.
2. Click the Delete button on the task bar.
3. Select "Yes" to confirm the deletion of the procedure output folder.

## Tracing Output

#### **Tip #6 – Tracing Procedure Output**

Output producing procedures often create multiple pieces or tables of information. In order to discriminate between the various pieces of information, it is advantageous to know the names assigned to each piece of information. The ability to display the names of individual pieces of information generated on output is referred to as **tracing**. The ODS statement syntax ODS trace ON / Listing; causes the SAS System to turn the trace feature on and print results to the SAS Listing destination.

```
ODS Trace ON / Listing;  
Proc univariate data=libref.movies;  
Run;  
ODS Trace Off;
```

The trace record displays information about the data component, the table definition, and the output object. For example, the trace record displays the following output objects to the SAS Listing destination: 1) Moments, 2) BasicMeasures, 3) TestForLocation, 4) Quantiles, and 5) ExtremeObs. A sample trace record containing each output object's name, label, template, and path is displayed for the Univariate procedure. Note that for each output object, the name, label, template, and path is displayed.

```
Output Added:  
-----  
Name: Moments  
Label: Moments  
Template: base.univariate.Moments  
Path: Univariate.age.Moments  
-----
```

Output Added:

-----  
Name: BasicMeasures  
Label: Basic Measures of Location and Variability  
Template: base.univariate.Measures  
Path: Univariate.age.BasicMeasures  
-----

Output Added:

-----  
Name: TestsForLocation  
Label: Tests For Location  
Template: base.univariate.Location  
Path: Univariate.age.TestsForLocation  
-----

Output Added:

-----  
Name: Quantiles  
Label: Quantiles  
Template: base.univariate.Quantiles  
Path: Univariate.age.Quantiles  
-----

Output Added:

-----  
Name: ExtremeObs  
Label: Extreme Observations  
Template: base.univariate.ExtObs  
Path: Univariate.age.ExtremeObs  
-----

## Selecting Output with ODS

A selection or exclusion list exists for each open ODS destination. These lists determine which output objects to send to ODS destinations. To accomplish this, ODS verifies whether an output object is included in a destination's selection or exclusion list. If it does not appear in this list, then the output object is not sent to the ODS destination. If an output object is included in the list, ODS determines if the object is included in the overall list. If it does not appear in this list, then the output object is not sent to the ODS destination. If an output object is included in the overall list then ODS sends it to the selected destination.

### **Tip #7 – Selecting Desired Pieces of Information**

Once you know the individual names of each output object (from the trace), you can then select the desired object for reporting purposes. The syntax is:

**ODS select *output-component-name*;**

where ***output-component-name*** is the name of the desired output object. To select just the output object Moments from the Univariate procedure, the following syntax is specified:

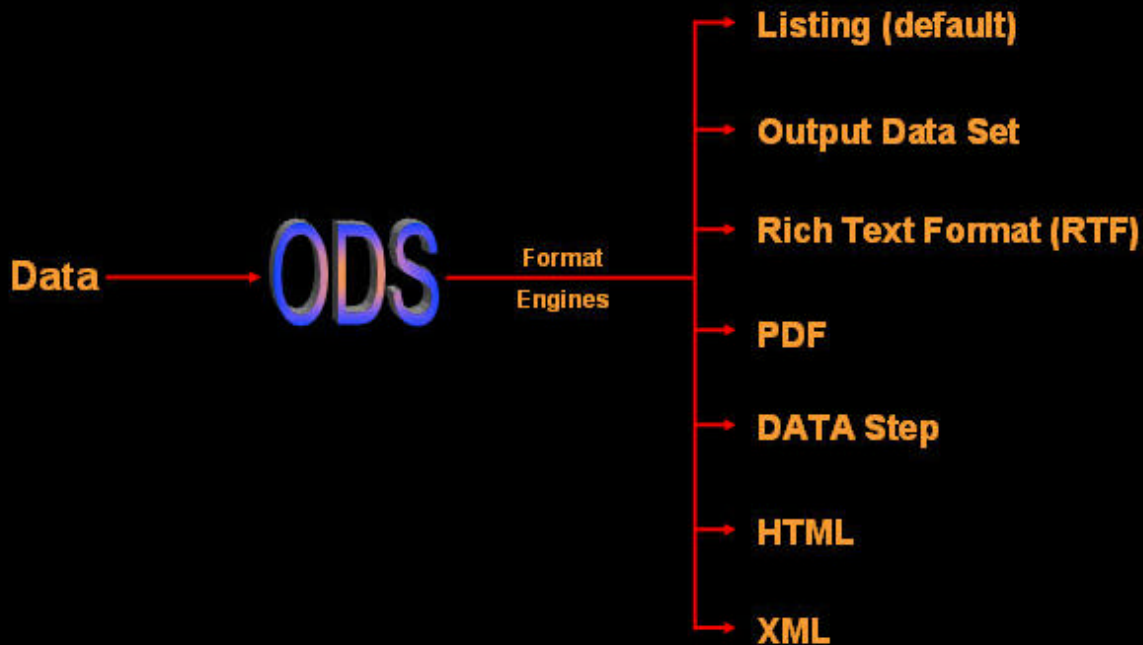
```
ODS Select Moments;  
Proc univariate data=libref.movies;  
Run;
```

## Creating Unique Output with ODS

Output Delivery System (ODS) can be used to create a variety of output formats. ODS statements are classified as global statements and are processed immediately by the SAS System. ODS statement options control what format engine(s) are turned on and in effect during the step or until another ODS statement is specified.

ODS has built-in format engines (e.g., Listing, Output, RTF, PDF, DATA Step, HTML, and XML). Specifying an ODS statement and destination at a particular point in a program is important, because output-producing PROC and DATA steps will respond by sending output to the open destination.

# Output Formats / Destinations



## Creating SAS Output Data Sets

### **Tip #8 – SAS Output Data Sets**

Occasionally, output results are needed in a SAS data set rather than in printed form such as the Listing destination. Re-directing SAS procedure output to a data set is relatively simple with ODS. The syntax is:

```
ODS Output output-table-name =  
      user-defined-table-name;  
< SAS Code >
```

where *output-table-name* is the name of the desired output table (component) containing the information you want written to a data set, such as *Moments* in the UNIVARIATE procedure. *User-defined-table-name* is the name you supply for the newly created data set. It can be defined as either a temporary or permanent data set. Once an object is selected, specify the object in the ODS OUTPUT statement. For example, the Moments from the Univariate procedure is selected and output to a SAS data set in the following code.

```
ODS Listing Close;  
ODS Output Moments = Movie_Moments;  
Proc univariate data=libref.movies;  
  Run;  
ODS Listing;
```

When the OUTPUT destination is no longer needed, it can be closed with the following ODS statement:

```
ODS OUTPUT CLOSE;
```

## Creating Rich Text Format (RTF)

### **Tip #9 – Rich Text Format (RTF)**

Rich text format (RTF) is text consisting of formatting attributes codes, such as boldface, italics, underline, etc. It is principally used to encapsulate text and formatting attributes during copy-and-paste operations. Because word-processing programs use RTF rather than ASCII when handling data, the need to reformat is a thing of the past. The syntax to create RTF output is:

```
ODS RTF FILE = 'user-specified-file-name';
```

where user-specified-file-name references a complete and fully-qualified output location for the creation and storage of the RTF file, data, and codes. For example, the following code creates an RTF file using the Univariate procedure output. (**Note:** The RTF extension is required).

```
ODS Select Moments = moments;  
ODS RTF FILE='ods-rtf-univariate.rtf';  
Proc univariate data=libref.movies;  
  Title1 "Delivering RTF Output";  
Run;  
ODS RTF Close;
```

The results of the RTF output are displayed below:

**Delivering RTF Output**  
**The UNIVARIATE Procedure**  
**Variable: Year**

Moments			
<b>N</b>	22	<b>Sum Weights</b>	22
<b>Mean</b>	1982.909 09	<b>Sum Observations</b>	43624
<b>Std Deviation</b>	15.21249 2	<b>Variance</b>	231.4199 13
<b>Skewness</b>	- 2.110647 3	<b>Kurtosis</b>	4.418072 09
<b>Uncorrected SS</b>	86507286	<b>Corrected SS</b>	4859.818 18
<b>Coeff Variation</b>	0.767180 51	<b>Std Error Mean</b>	3.243314 2

## Creating Postscript Output

### **Tip #10 – Creating and Naming Postscript Files**

ODS enables the creation of a postscript file from SAS output to help ensure that formatting and content is preserved while maintaining complete printer independence. The next example shows the creation of a postscript file from Print procedure output.

```
ODS Listing Close;  
ODS Printer Postscript;  
Proc print data=libref.movies noobs n;  
  Title1  
  "Example Creating a Postscript File";  
Run;  
ODS Printer Close;  
ODS Listing;
```

Since the postscript file was created without assigning a filename, ODS assigns the name: **SASPRT.PS**. Frequently, a user-defined filename is desired rather than the default name. In these cases, the ODS statement and **File=** option can be used to assign a name to the postscript file.

```

ODS Listing Close;
ODS Printer Postscript
  File='Beginning_tutorials.ps';
  Proc print data=libref.movies noobs n;
  Title1
  "Example Creating a Postscript File";
  Run;
ODS Printer Close;
ODS Listing;

```

## ODS and the DATA Step

### Tip #11 – The Custom Reporting Interface

To provide greater capabilities when working with custom output created in the DATA step, ODS is integrated into the DATA step. Two new options are necessary to take advantage of ODS: 1) the ODS option in the FILE statement and 2) the **\_ODS\_** option in the PUT statement. These two options are used to direct the results of a DATA step to ODS. For example, the following code directs output from a DATA step through the RTF format engine to create a RTF file.

```

ODS RTF FILE='ods-DATA-step.rtf';
  Title1 "Great Movies";
  Data NULL;
  Set ods.movies;
  File print ods;
  Put _ods_;
  Run;
ODS RTF Close;

```

The RTF file created in the previous DATA step code is illustrated.

### *Great Movies*

Title	Length	Category	Year	Studio	Rating
Brave Heart	177	Action Adventure	1995	Paramount Pictures	R
Casablanca	103	Drama	1942	MGM / UA	PG
Christmas Vacation	97	Comedy	1989	Warner Brothers	PG-13
Coming to America	116	Comedy	1988	Paramount Pictures	R
Dracula	130	Horror	1993	Columbia TriStar	R
Dressed to Kill	105	Drama Mysteries	1980	Filmways Pictures	R
Forrest Gump	142	Drama	1994	Paramount Pictures	PG-13
Ghost	127	Drama Romance	1990	Paramount Pictures	PG-13
Jaws	125	Action Adventure	1975	Universal Studios	PG
Jurassic Park	127	Action	1993	Universal Pictures	PG-13
Lethal Weapon	110	Action Cops & Robber	1987	Warner Brothers	R
Michael	106	Drama	1997	Warner Brothers	PG-13
National Lampoon's Vacation	98	Comedy	1983	Warner Brothers	PG-13
Poltergeist	115	Horror	1982	MGM / UA	PG
Rocky	120	Action Adventure	1976	MGM / UA	PG

## Output Delivery Goes Web

The Web offers incredible potential that impacts all corners of society. With its increasing popularity as a communications medium, Web publishers have arguably established the Web as the greatest medium ever created. Businesses, government agencies, professional associations, schools, libraries, research agencies, and a potpourri of society's true believers have endorsed the Web as an efficient means of conveying their messages to the world.

The Web is not a static environment, but a dynamic medium capable of distributing content anywhere and at anytime. The 24/7 model permits information to be refreshed and updated continuously as new material becomes available. A primary objective of Web publishers everywhere is to engage visitors with timely and interesting content that brings them back for more.

### **Tip #12 – Pagesize / Linesize Settings**

The Options PS= and LS= have no effect when used with the HTML destination (opposed to most other output-producing steps that generate output to a print destination). If the PS= and/or LS= options are used with the HTML destination, they are simply ignored. The SAS System creates a type of “streaming” or continuous output and adds elevator bars (horizontal and/or vertical) for easy navigation.

The SAS System does provide a way for users to paginate through output displayed in a body file. The HTML destination provides a way to designate an optional description of each page of the body file. The PAGE= file (when specified) recognizes each new page of output produced by ODS. What ODS does is create a section called **Table of Pages** containing links to the body file for easy navigation through output.

### **Tip #13 – Deploying Output to the Web**

With the popularity of the Internet, you may find it useful to deploy selected pieces of output on your web site. ODS makes deploying procedure output to the web a simple process. Syntactically correct HTML code is automatically produced and made ready for deployment using one of the Internet browser software products (e.g., Internet Explorer, Netscape Navigator, etc.). As a result, the SAS System and the HTML destination create a type of “streaming” or continuous output by adding elevator bars (horizontal and/or vertical) for easy navigation.

### **Tip #14 – HTML Destination Files**

Four types of files can be created with the ODS HTML destination: 1) body, 2) contents, 3) page, and 4) frame. Each file is described below.

The **Body** file contains the results from the procedure embedded in ODS-generated HTML code. Horizontal and vertical scroll bars are automatically placed on the generated page, if necessary.

The **Contents** file consists of a link to each HTML table within the body file. It uses an anchor tag to link to each table. By using your browser software, you can view the contents file directly or as part of the frame file.

The **Page** file consists of a link to each page of ODS created output. By using your browser, you can view the page file directly or as part of the frame file.

The **Frame** file displays the body file and the contents file, the page file, or both. The next example shows the creation of Web-ready Univariate procedure output using the HTML format engine with the body=, contents=, page=, and frame= options.

```
ODS Listing Close;
ODS HTML body='ods-body.htm'
      contents='ods-contents.htm'
      page='ods-page.htm'
      frame='ods-frame.htm';
proc univariate data=libref.movies;
  Title1 'Creating HTML Output with ODS';
run;
ODS HTML Close;
ODS Listing;
```

A snippet of the HTML output appears on the next page:

Creating HTML Output with ODS HTML FRAME File containing UNIVARIATE Ou			
The UNIVARIATE Procedure Variable: Length			
Moments			
<b>N</b>	22	<b>Sum Weights</b>	
<b>Mean</b>	124.909091	<b>Sum Observations</b>	27
<b>Std Deviation</b>	25.8344714	<b>Variance</b>	667.4199
<b>Skewness</b>	1.45414494	<b>Kurtosis</b>	1.714538
<b>Uncorrected SS</b>	357266	<b>Corrected SS</b>	14015.81
<b>Coeff Variation</b>	20.682619	<b>Std Error Mean</b>	5.507927

### **Tip #15 – 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 from the UNIVARIATE procedure, the ODS PDF option can be 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 PDF Close;
ODS Listing;
```

### **Tip #16 – Combining Output Results**

With the streaming capabilities of HTML output, results can be combined so they appear on the same screen (or page). 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**

Table of Contents	
1. The Print Procedure ·Data Set <a href="#">ODSLIB.MOVIES</a>	
2. The Means Procedure ·Summary statistics	
Table of Pages	
1. The Print Procedure ·Page 1	
2. The Means Procedure ·Page 2	

**Movie Classics Listing**

Title	Length	Category	Year	Studio	Rating
The Wizard of Oz	101	Adventure	1939	MGM / UA	G
Casablanca	103	Drama	1942	MGM / UA	PG
Jaws	125	Action Adventure	1975	Universal Studios	PG
Rocky	120	Action Adventure	1976	MGM / UA	PG
Star Wars	124	Action Sci-Fi	1977	Lucas Film Ltd	PG
Poltergeist	115	Horror	1982	MGM / UA	PG
The Hunt for Red October	135	Action Adventure	1989	Paramount Pictures	PG
N = 7					

**Summary of Movie Classics**

**The MEANS Procedure**

Rating	N Obs	Variable	N	Mean	Std Dev	Minimum	Maximum
G	1	Length	1	101.0000000	.	101.0000000	101.0000000
		Year	1	1939.00	.	1939.00	1939.00
PG	6	Length	6	120.3333333	10.7641380	103.0000000	135.0000000
		Year	6	1973.50	16.2818918	1942.00	1989.00
PG-13	7	Length	7	127.2857143	33.9004144	97.0000000	194.0000000
		Year	7	1991.86	4.9809160	1983.00	1997.00
R	8	Length	8	129.2500000	28.4190580	105.0000000	177.0000000
		Year	8	1987.63	5.1806646	1980.00	1995.00

**Tip #17 – Testing Web Output**

Before transferring your Web files to a Web server, users should thoroughly test any HTML code to make sure they are problem-free. Many viewers will not return to Web pages that contain errors or do not work according to design. A word of caution: **Not all Web browser software handles web pages the same way.** Microsoft Internet Explorer may display your web pages differently than Netscape Navigator and others.

Before deploying web-based output to the Web or Intranet, it is important to visually inspect and test your output to see how your Web pages behave. Web validation services are available for users to check web pages for errors or inconsistencies. The following recommendations should be considered before deploying Web output.

1. Check spelling – check the spelling on each of your Web pages before making them available.
2. If possible, use a validation service to identify errors in your use of HTML.
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 enables 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 sound 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 a monitor can display.

## Building 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.

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.

### **Tip #18 – Building a Drill-down Application**

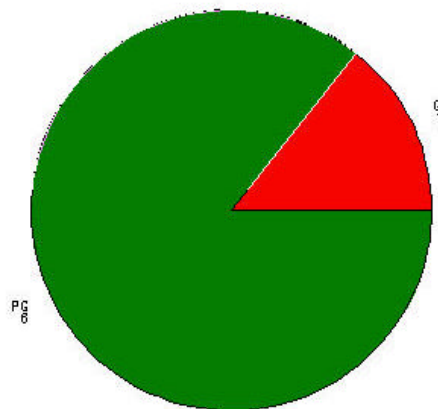
To simplify the process of building a graphical drill-down application in the SAS System, five easy steps are presented.

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.

Drill-down Graphical Univariate Application  
FREQUENCY of Rating



<A HREF="drill-down-graphical-univariate-application.html">

Display Graph</A>

## Example Code

The SAS coding steps used in building 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;
  title "Drill-down Graphical Application";
  pie rating / html=urllink;
  where upcase(rating) in ('G', 'PG');
run;
quit;

*STEP 4a - 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 4b - 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;
```

## Conclusion

The time has come to turn tired-looking output listings into great looking information. The SAS® Output Delivery System (**ODS**) provides new and exciting ways to improve the way output is generated. Gone are the days when the only available formatting choice for output was basic output listings consisting of monospace fonts printed on a white background.

ODS' built-in format engines addresses the inherent weaknesses found in traditional SAS output by enabling "quality" output to be produced without the need of importing it into word processors. The latest version boasts many new format destinations including Postscript, RTF, HTML, SAS data sets, and DATA step interfaces, while supporting traditional Listing output as well. Users have a powerful and easy way to create and access formatted procedure and DATA step output.

## References

- Gupta, Sunil Kumar (2003) , Quick Results with the Output Delivery System (Art Carpenter's SAS Software), SAS Institute Inc., Cary, NC, USA.
- Heffner, William F. (1998), "ODS: The DATA Step Knows," Proceedings of the 23<sup>rd</sup> Annual SAS Users Group International Conference, Cary, NC: SAS Institute Inc.
- Lafler, Kirk Paul (2006), "Output Delivery System Tips and Techniques," Proceedings of the 2006 South Central SAS

- Users (SCSUG) Conference, Software Intelligence Corporation, Spring Valley, CA, USA
- Lafler, Kirk Paul (2004), "Creating HTML Output with Output Delivery System," Proceedings of the 2004 Western Users of SAS Software (WUSS) Conference, Software Intelligence Corporation, Spring Valley, CA, USA
- Lafler, Kirk Paul (2004), "Output Delivery Tips, Tricks, and Techniques," Proceedings of the 2004 North Texas SAS Users Group International Conference, Software Intelligence Corporation, Spring Valley, CA, USA.
- Lafler, Kirk Paul (2002), "Output Delivery Goes Web," Proceedings of the 27<sup>th</sup> Annual SAS Users Group International Conference, Software Intelligence Corporation, Spring Valley, CA, USA.
- Lafler, Kirk Paul (2000), The SAS<sup>®</sup> 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 24<sup>th</sup> 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<sup>®</sup> 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<sup>®</sup> 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.

## Acknowledgments

The author would like to thank John Xu and Delayne.H.Stokke, MWSUG Co-Chairs, for asking me to present at and to prepare this paper for MWSUG 2007. Thank you for a great conference!

## Trademark Citations

SAS, SAS Alliance Partner, and SAS Certified Professional are registered trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

## About the Author

Kirk Paul Lafler is consultant and founder of Software Intelligence Corporation and has been programming in SAS since 1979. As a SAS Certified Professional and SAS Institute Alliance Member (1996 – 2002), Kirk provides IT consulting services and training to SAS users around the world. As the author of four books including *PROC SQL: Beyond the Basics Using SAS* (SAS Institute, 2004), Kirk has written more than two hundred peer-reviewed papers and articles that have appeared in professional journals and SAS User Group proceedings. He has also been an invited speaker at more than two hundred SAS International, regional, local, and special-interest user group conferences and meetings throughout North America. His popular SAS Tips column, "Kirk's Korner of Quick and Simple Tips", appears regularly in several SAS User Group newsletters and Web sites, and his fun-filled SASword Puzzles is featured in SAScommunity.org. Comments and suggestions can be sent to:

Kirk Paul Lafler  
Software Intelligence Corporation  
World Headquarters  
P.O. Box 1390  
Spring Valley, California 91979-1390  
E-mail: KirkLafler@cs.com

