Paper 58-27

Quick Results with the Output Delivery System

Sunil K. Gupta, Gupta Programming, Simi Valley, CA

ABSTRACT

SAS®'s new Output Delivery System (ODS) opens a whole new world of options in generating quality, detailed presentation output. ODS enables the creation of various new file types including Rich Text Format (RTF), PDF, PostScript, HTML and SAS data sets. SAS users can take advantage of ODS features to easily convert existing SAS code into these new file types. By having a selection of styles available to choose from, SAS users can control the report's overall color scheme, font and size.

This paper introduces the concepts necessary to understand and apply the basic features of SAS's ODS. Issues in defining and selecting output destinations, selecting output objects and creating output files will be discussed. The focus is on the creation of various types of output files through step-by-step examples.

INTRODUCTION

SAS's new ODS features offer significant improvements in the presentation of reports and files. Some of ODS's advantages include the following:
➢ Creation of output objects from most all procedures
➢ Creation of RTF, HTML and PostScript files
➢ Creation of data sets from output objects

The advanced features of ODS enable greater control and flexibility for the presentation of the data. Some of ODS's advanced features include the following:
➢ Selection of "SAS-Supplied" Styles
➢ Support for web site navigation and management

This paper will show how to apply the basic features of ODS to create HTML and RTF files along with SAS Data Sets. In addition, options to control features such as styles and web site management will be demonstrated.

ODS BASICS AND PROCESSES

The sequence of steps to follow in using ODS for file generation include the following:
1. Creating Output Files
2. Creating Output Files with Style
3. Identifying Output Objects
4. Selecting Output Objects

When working in the ODS environment, it is helpful to consider the following definitions: (Note these are not SAS Institute's official definitions.)

1. DESTINATION – "Final File Type"
Where you want to be?

(List, HTML, RTF, Printer, PDF, Data set)

2. OBJECT – "A Non-Physical Item"
What you have to work with?
(Select or Exclude Output Objects)

3. STANDARD REPORT – "As-Is Final End Product"
How you will reach your destination?
(Default Attributes defined in Default Style & Template)
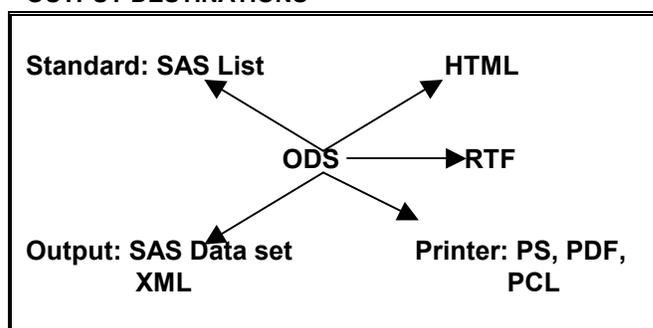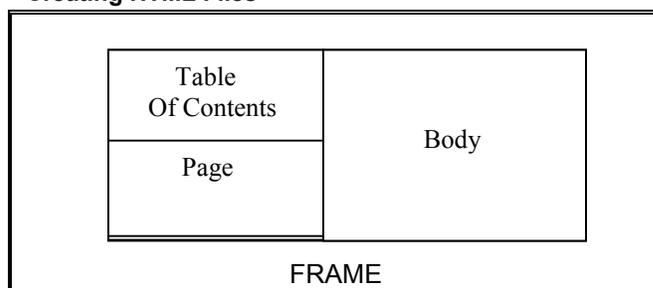
1. CREATING OUTPUT FILES

In creating output files, ODS needs to open a destination to use and then close that output destination after the SAS procedure that generates the results. ODS categorizes destinations by Report and Data types.

Currently, the available Report output destinations include:
➢ Standard SAS List
➢ HTML
➢ RTF
➢ Printer – PostScript, PDF, PCL

Currently, the available Data output destinations include:
➢ Output - SAS Data Set, XML

OUTPUT DESTINATIONS


```
Standard: SAS List              HTML



                    ODS ——→ RTF



Output: SAS Data set        Printer: PS, PDF,
         XML                           PCL
```

Creating HTML Files


```
        Table
      Of Contents            Body

        Page

              FRAME
```

Opening the HTML destination is accomplished with the ODS HTML statement. The HTML file destination has options to save the content into several logical files. The advantage of creating all of these HTML files is to facilitate navigation and the management of the web site. ODS automatically creates the links to each section of the body. Note that the BODY = option can be replaced with the FILE = option. After PROC UNIVARIATE, the ODS HTML CLOSE statement saves the results of PROC UNIVARIATE to the HTML file.

```
 ODS HTML
   PATH = 'PATH-spec'         (folder for html files)
   BODY = 'HTML-FILE-spec' (html filename)
   CONTENTS = 'TOC-spec'    (links to pieces in body)
   PAGE = 'PAGE-spec'         (individual pages)
   FRAME = 'FRAME-spec'     (integrate toc, body &
                                          pages);
```

```
* Example: Create HTML file;

ODS HTML
BODY = 'c:\sasgroup\SUGI27\demog_body.html';

PROC UNIVARIATE DATA=DEMOG;
  VAR WEIGHT;
RUN;

ODS HTML CLOSE;
```


HTML File

The next example shows how the results of PROC FREQ and PROC UNIVARIATE are all saved to one set of HTML files. All the results from each SAS procedure after the ODS HTML statement and before the ODS HTML CLOSE statement are saved to the HTML files.

```
* Example: Create HTML file of several SAS procedures;

ODS HTML
PATH = 'c:\sasgroup\SUGI27\' (url=none)
          BODY = 'body.html'
          CONTENTS = 'toc.html'
          FRAME = 'frame.html' ;

PROC FREQ DATA=DEMOG;
```

```
 TABLES DRUG;
RUN;

PROC UNIVARIATE DATA=DEMOG;
 VAR WEIGHT;
RUN;

ODS HTML CLOSE;
```

HTML Files with several SAS procedures

Creating Output SAS Data Sets

With ODS, the results of any SAS procedure can be saved as an Output data set. Note that in order to use the ODS OUTPUT statement, the object name or the object path must be specified. The syntax for creating an output data set is to set the object name (BASICMEASURES) equal to the new data set name (MEASURE).

```
* Example: Create Output Data Set;

ODS OUTPUT BASICMEASURES = MEASURE;

PROC UNIVARIATE DATA=DEMOG;
 VAR WEIGHT;
RUN;

ODS OUTPUT CLOSE;
```

```
* Proc contents of MEASURE Output Data Set;
#    Variable       Type    Len    Pos     Format
————————————————————————————————————————————————
2    LocMeasure     Char     6      22
3    LocValue       Num      8       0     8.4
4    VarMeasure     Char    19      28
1    VarName        Char     6      16
5    VarValue       Num      8       8     D10.

* Proc print of MEASURE Output Data Set;
         Measure as SAS Data set

      Var       Loc
Obs   Name    Measure LocValue  VarMeasure    VarValue

 1   weight  Mean    198.6800  Std Deviation
44.55682
 2   weight  Median 199.0000  Variance
1985
 3   weight  Mode       .       Range
```

```
159.00000
 4   weight            _         Interquartile Range
56.00000
```

Creating HTML and RTF Files

The process of creating more than one output file is similar to that of creating a single output file. An additional ODS statement must be added for each destination before the SAS procedure that generates the results. In addition, another ODS CLOSE statement must be added at the end to create the output file.

```
* Example: Create HTML and RTF files;

ODS HTML
FILE = 'c:\sasgroup\SUGI27\demog.html' ;

ODS RTF FILE= 'c:\sasgroup\SUGI27\'demog.rtf' ;


PROC UNIVARIATE DATA=DEMOG;
 VAR WEIGHT;
RUN;

ODS RTF CLOSE;

ODS HTML CLOSE;
```
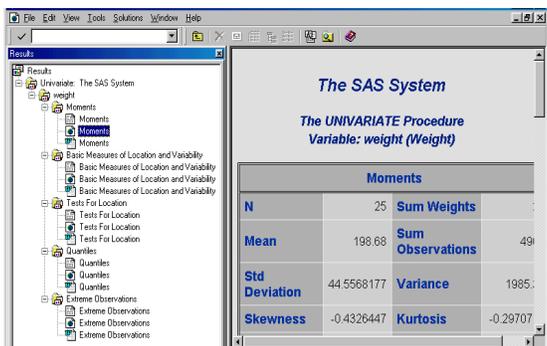
As you can see in the SAS Display Manager, the LISTING, HTML and RTF files are created. The information contained in each file is the same. Note that the LISTING file is generated because it is the default output destination that is opened until closed.

SAS Display Manager

2. CREATING OUTPUT FILES WITH STYLE

Another key feature of using ODS to create reports is to take advantage of using styles. Styles enhance the overall look of the report with colors, font and size. "SAS-Supplied" are included during the installation and can be used with ODS for more visually appearing output.

Note that styles, however, are supported by destinations that support report details such as color, font and size.

STYLE SUPPORT BY DESTINATION

DESTINATION	SUPPORT STYLE
Listing	No
Printer	Yes
RTF	Yes
Data Set	No
HTML	Yes

The two options for creating reports are to use the default style or to specify a different style. To send output to a destination file using a "SAS-Supplied" style, use the STYLE = option.

ODS *DESTINATION*   FILE="file-spec.ext"
                     STYLE = style-spec;

Where *DESTINATION* is one of the following: HTML, RTF or Printer.
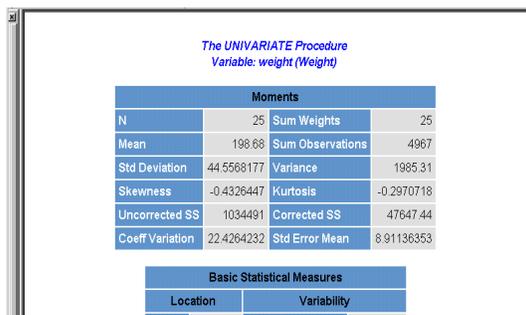
```
* Example: Create HTML file with SAS supplied style;

ODS HTML FILE='c:\sasgroup\SUGI27\demog_style.html'
        STYLE=barrettsblue;

PROC UNIVARIATE DATA=DEMOG;
 VAR WEIGHT;
RUN;

ODS HTML CLOSE;
```

HTML File with Style

Below is a collection of existing styles that can be used with the STYLE= option. Note that the default style for the LISTING and the HTML destinations is different from the default style for the PRINTER and the RTF destinations. In addition, there are several styles that are most appropriate for specific destinations such as the Printer.

"SAS Supplied" STYLES

NAME	DESCRIPTION
BarrettsBlue	Blue header background, light table background
Beige	Beige header text, white text in table
Brick	Brick color header text, white text in table
Brown	Brown title, black header, light table

3

	background
D3D	White header, bold table border
Default	Dark blue header, shade table background (Default for LISTING and HTML Destinations)
Minimal	No color, light text in table
NoFontDefault	Black header text, white background table
Printer	Printer Style (Default for PRINTER Destination)
RTF	RTF Style (Default for RTF Destination)
Statdoc	Blue header, black text in table
Theme	Dark header, dark table
FancyPrinter	Printer Style
SansPrinter	Printer Style
SasdocPrinter	Printer Style
SerifPrinter	Printer Style

To understand how styles affect the output file, it is helpful to consider the output file as a composite of global attributes defined by the style template. Styles define the overall attributes of the output file such as color, font face and size. The report style defines the overall look of the output.

The fonts used in styles define items such as titles and headers. The font definition consists of the following items: font face, font size, font weight, font style, font width.

Examples of the font face include the following: times, courier, arial, Helvetica.

FONT STYLE ELEMENT

FONT WEIGHT	FONT STYLE	FONT WIDTH
Medium	Italic	Normal
Bold	Roman	Compressed
Demi_bold	Slant	Extra_compressed
Extra_bold		Narrow
Light		Wide

The colors to be used in the report can be any color that is supported by SAS/GRAPH®.

3. IDENTIFYING OUTPUT OBJECTS

ODS represents a new way in which SAS creates objects to store data from each SAS procedure. ODS provides tools to help identify objects. For the purpose of restricting output to the report, it is essential to know the objects to select.

Identifying Output Objects

The method used to identify objects from the SAS Log is the ODS TRACE statement. Use the ODS TRACE statement to write a record of each output object that is created to the SAS Log. Including the LABEL and LISTING options helps to provide a label of the object

along with displaying the object information just before the procedure results in the SAS List file.

```
* Example: Use ODS TRACE statement.

ODS TRACE ON / LABEL LISTING;

PROC UNIVARIATE DATA=DEMOG;
 VAR WEIGHT;
RUN;

ODS TRACE OFF;
```

PARTIAL SAS LIST OF ODS TRACE STATEMENT

```
Output Added:
-------------
Name:       BasicMeasures
Label:      Basic Measures of Location and
Variability
Template:   base.univariate.Measures
Path:       Univariate.weight.BasicMeasures
Label Path: 'The Univariate
Procedure'.'weight'.'Basic Measures
of Location and Variability'
-------------
```

OUTPUT OBJECT ATTRIBUTES

ATTRIBUTE	DESCRIPTION
NAME	Output object name
DATA	Data component used to create output object (Note: Appears only if different from name of output object)
LABEL	Contents of the output object
TEMPLATE	Template component used to format the object
PATH	Path of the output object

ODS creates output objects from the execution of each SAS procedure. Procedure output is divided into one or more output objects. Each output object has a set of attributes such as name and label.

Each output object has two components.
1) Data component   (raw numbers and characters)
2) Template component   (description of format and arrangement instructions)

OUTPUT OBJECT

Data Component

Template Component
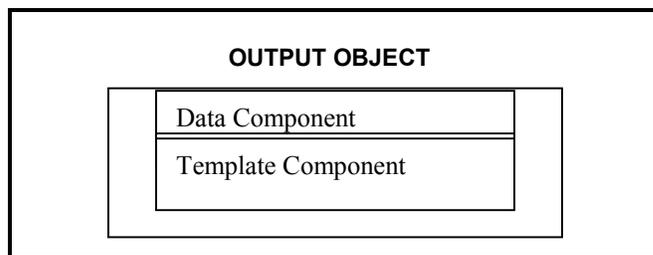
4. SELECTING OUTPUT OBJECTS

To restrict information to the output file, you need to select the output object created by ODS. ODS enables users to select objects by several ways. This paper will review the following two methods to select objects: by object name and by object path.

Selecting Output Objects by Object Name

Selecting objects to include in the output file is accomplished with the SELECT option. Use the ODS SELECT option to select several objects (name1 name2 … namen, where name1, 2, n are the object's name). The SHOW option can be used to verify the selection list. Only the information in the BASICMEASURES object will be included in the HTML file.

```
* Example: Use SELECT option to restrict information.

ODS HTML  FILE =
'c:\sasgroup\sugi27\select_demog.html' ;

ODS HTML SELECT BASICMEASURES;
ODS HTML SHOW;

 PROC UNIVARIATE DATA=DEMOG;
 VAR WEIGHT;
 RUN;

ODS HTML CLOSE;
```
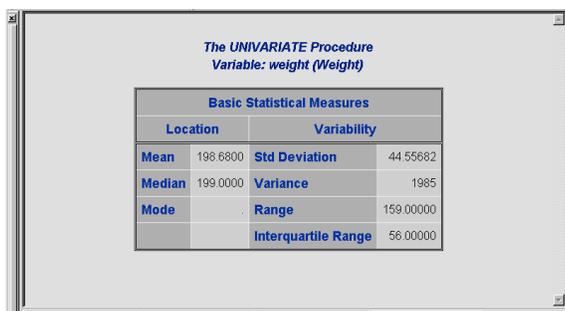
PARTIAL SAS LOG OF SELECT OPTION

```
61   ods html select basicmeasures;
62   ods html show;
Current HTML select list is:
1. basicmeasures
63   proc univariate data=demog;
64    var weight;
65   run;


NOTE: PROCEDURE UNIVARIATE used:
     real time           0.05 seconds
```

HTML FILE

Selecting Output Objects by Object Path

An alternative to referencing by object name is to use the object's path name. In this case, the same object is selected. Use the ODS SELECT option to select by path.

```
* Example: Use SELECT option to restrict information.

ODS HTML  FILE =
'c:\sasgroup\sugi27\select_demog.html' ;

ODS HTML SELECT
UNIVARIATE.WEIGHT.BASICMEASURES;
ODS HTML SHOW;

PROC UNIVARIATE DATA=DEMOG;
  VAR WEIGHT;
RUN;

ODS HTML CLOSE;
```
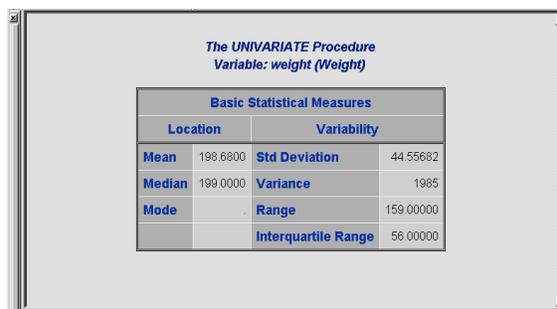
PARTIAL SAS LOG OF SELECT OPTION

```
68   ods html select
univariate.weight.basicmeasures;
69   ods html show;
Current HTML select list is:
1. univariate.weight.basicmeasures
70   proc univariate data=demog;
71    var weight;
72   run;


NOTE: PROCEDURE UNIVARIATE used:
     real time           0.10 seconds
```

HTML FILE

Selecting Multiple Output Objects from By Statements

When using the by statement, ODS creates multiple objects from the execution of the SAS procedure. Since multiple objects are created, it is better to select objects by the object's path name. In the DEMOG data set, there are two drugs used in the study. Due to the by statement, objects will be created for each drug group. In this example, the BASICMEASURES information is now provided by each drug group.

```
* Example: Use SELECT option with path name to restrict
information.

PROC SORT DATA=DEMOG;
 BY DRUG;
RUN;

ODS HTML
FILE='c:\sasgroup\sugi27\select_weight_demog';

ODS SELECT
UNIVARIATE.BYGROUP1.WEIGHT.BASICMEASURES
UNIVARIATE.BYGROUP2.WEIGHT.BASICMEASURES
;
PROC UNIVARIATE DATA=DEMOG;
 BY DRUG;
 VAR WEIGHT;
RUN;

ODS HTML CLOSE;
```

HTML FILE

Selecting Multiple Output Objects from Multiple Analysis Variables

When specifying multiple variables, ODS creates multiple objects from the execution of the SAS procedure. Since multiple objects are created, it is better to select objects by the object's path name. In this example, the BASICMEASURES of both weight and height are saved to the HTML file.

```
* Example: Use SELECT option with path name to restrict
information.

ODS HTML
FILE='c:\sasgroup\sugi27\select_weight_height_demog';

ODS SELECT UNIVARIATE.WEIGHT.BASICMEASURES
UNIVARIATE.HEIGHT.BASICMEASURES;

PROC UNIVARIATE DATA=DEMOG;
 VAR WEIGHT HEIGHT;
RUN;

ODS HTML CLOSE;
```

HTML FILE

Excluding Output Objects

An alternative to selecting objects is to exclude objects. Excluding objects from the output file is accomplished with the EXCLUDE option. Use the ODS EXCLUDE option to exclude an object. In this example, all objects except the BASICMEASURES object will be included in the HTML file.

```
* Example: Use EXCLUDE option to restrict information.

ODS HTML FILE =
'c:\sasgroup\sugi27\exclude_demog.html'

ODS HTML EXCLUDE BASICMEASURES;

ODS HTML SHOW;

PROC UNIVARIATE DATA=DEMOG;
 VAR WEIGHT;
RUN;

ODS HTML CLOSE;
```

PARTIAL SAS LOG OF EXCLUDE OPTION

```
117  ods html exclude basicmeasures;
118  ods html show;
Current HTML exclude list is:
1. basicmeasures
119  proc univariate data=demog;
120   var weight;
121  run;
```

Maintaining Constant Selection List

The behavior of ODS is to reset the selection list after each SAS procedure. This could be a problem if you want to keep the selection list constant. To have a constant selection list across boundaries, use the PERSIST option to maintain the selection of objects after each SAS procedure.

```
* Example: Use SELECT option to restrict information.

ODS HTML  FILE =
'c:\sasgroup\sugi27\select_persist_demog.html' ;

ODS HTML SELECT BASICMEASURES (PERSIST);
ODS HTML SHOW;

TITLE 'RUN 1';
PROC UNIVARIATE DATA=DEMOG;
 VAR WEIGHT;
RUN;

ODS HTML SHOW;

TITLE 'RUN 2';
PROC UNIVARIATE DATA=DEMOG;
 VAR WEIGHT;
RUN;

ODS HTML CLOSE;
```

PARTIAL SAS LOG OF SELECT OPTION

```
125  ods html select basicmeasures (persist);
126  ods html show;
Current HTML select list is:
1. basicmeasures(PERSIST)
127  title 'Run 1';
128  proc univariate data=demog;
129   var weight;
130  run;


NOTE: PROCEDURE UNIVARIATE used:
      real time            0.10 seconds


131  ods html show;
Current HTML select list is:
1. basicmeasures(PERSIST)
132  title 'Run 2';
133  proc univariate data=demog;
134   var weight;
135  run;


NOTE: PROCEDURE UNIVARIATE used:
      real time            0.04 seconds
```

Multiple Objects to Different Destinations

Another key feature of ODS is the ability to simultaneously create multiple files containing different objects. Since the objects are independent of the destinations, it is possible for ODS to select different objects for different destinations. This enables great flexibility with ODS for file generation and content. In this example, the BASICMEASURES object is selected for the HTML file and the EXTREMEOBS object is selected for the RTF file.

```
* Example: Use SELECT option to restrict information.
```

```
ODS HTML
FILE = 'c:\sasgroup\sugi27\basic_demog.html' ;
ODS HTML SELECT BASICMEASURES;

ODS RTF FILE= 'c:\sasgroup\sugi27\extreme_demog.rtf' ;
ODS RTF SELECT EXTREMEOBS;

PROC UNIVARIATE DATA=DEMOG;
 VAR WEIGHT;
RUN;

ODS RTF CLOSE;
ODS HTML CLOSE;
```

HTML FILE

RTF FILE

Below is an object reference table for selected SAS procedures. It can be helpful to identify the object to specify in the ODS SELECT option.

OBJECT REFERENCE BY SAS PROCEDURE

PROCEDURE	NAME	PATH
Proc_name;	Obj_name	Proc_name. X.Obj_name
Var X;		
Run;		
Freq	OneWay Freqs	Freq.X. OneWayFreqs
Means	Summary	Means. Summary
SQL	SQL_ Results	SQL.SQL_ Results

Univariate	Moments	Univariate.X.Moments
	Basic Measures	Univariate.X.Measures
	TestsFor Location	Univariate.X. Location
	TestsFor Normal.	Univariate.X.Normality
	Quantilies	Univariate.X.Quantiles
	Extreme.	Univariate.X.ExtObs
	Frequency	Univariate.X.Frequency
	Plots	Univariate.X.Plots

SUMMARY

SAS's new Output Delivery System (ODS) feature enables the creation of various new file types including Rich Text Format (RTF), PostScript and HTML. In addition, creating SAS data sets from most any SAS procedure provides a standard method to extract information. Great power and flexibility is provided with ODS to create high quality output.

TRADEMARK INFORMATION

SAS® is a registered trademark of the SAS Institute Inc., Cary, NC, USA.

REFERENCES

WEB SITES
http://www.sas.com
http://www.sas.com/rnd/base/

SUGI PAPERS
Fehlner, William, "Making the Output Delivery System (ODS) Work for You", SAS Institute (Canada) Inc., Toronto, Ontario

Gupta, Sunil, "Customized Reports with SAS ODS", WUSS 2000 Class, Gupta Programming, Simi Valley, CA

Lafler, Kirk Paul, "Creating HTML Output with Output Delivery System", SUGI 25, Software Intelligence Corporation, Spring Valley, CA

Lafler, Kirk Paul, "Delivering Results with the Output Delivery System", SUGI 24, Software Intelligence Corporation, Spring Valley, CA

McNeill, Sandy, "ODS for Dummies", WUSS 2000, SAS Institute Inc., Cary, NC

Olinger, Chris, "ODS for Dummies", SUGI 25, SAS Institute Inc., Cary, NC

ABOUT THE AUTHOR

The author welcomes your comments & suggestions.

Sunil K. Gupta

Gupta Programming
SAS Institute Quality Partner™
213 Goldenwood Circle, Simi Valley, CA 93065
Phone: (805)-577-8877
E-mail: Sunil@GuptaProgramming.com
http://www.GuptaProgramming.com

Sunil is a senior consultant at Gupta Programming. He specializes in SAS/BASE®, SAS/AF®, SAS/FSP®, SAS/STAT® and SAS/GRAPH®. His consulting projects with pharmaceutical companies include the development of a Clinical Study Data Entry System, a Macro-Based Application for Report Generation, and customized plots and charts with SAS/GRAPH®. He has been using SAS® software for over 10 years and is a SAS Certified Professional V6™. He is also the author of a Books By User book on the Output Delivery System and was a SAS Institute Quality Partner™ for over 5 years.