

## Essential ODS Techniques for Creating Reports in PDF

Patrick Thornton, SRI International, Menlo Park, CA

### ABSTRACT

This paper lists and briefly describes ODS techniques that I have found most useful in capturing SAS 9.1 results and creating user friendly PDF reports on a Windows operating system. The techniques are presented for the beginning SAS user with examples and references to more detailed papers. The ODS techniques include capturing one-way frequency output to a data set, creating a PDF document, changing styles, controlling page setup (e.g. orientation, date, title, and footnote), using PUT, and formatting and traffic-lighting PROC REPORT.

### INTRODUCTION

The purpose of this paper is to list and briefly describe the most important techniques I have used while creating stakeholder-friendly PDF reports. Table 1 provides an overview where the purpose for each row's example is stated in column 1, the technique described in column 2, an example in column 3 and the result in the remaining column.

TABLE 1. OVERVIEW OF ESSENTIAL ODS TECHNIQUES

| Purpose                                  | ODS Technique                                                                                                                                       | Example                                                                                                                                                                          | Example Result                                                                                                                                                   |
|------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Determine data available through ODS     | Using ODS send data object information to the LOG by specifying TRACE ON.                                                                           | <pre>ods trace on; proc freq data=mydata;   tables _all_; run; ods trace off;</pre>                                                                                              | The name, label, template and path were displayed for each data object. The name ONEWAYFREQS was determined.                                                     |
| Obtaining data                           | Using ODS OUTPUT specify the object name and data set to be created.<br><b>objectname=dataset</b>                                                   | <pre>ods output onewayfreqs=freqs; proc freq data=mydata;   tables _all_; run; ods output close;</pre>                                                                           | Data set FREQS was created with the count and percent for all variables from the data set MYDATA.                                                                |
| Creating a PDF                           | Using ODS PDF specify the full path of a PDF.<br><b>file=full_path</b>                                                                              | <pre>ods pdf file='c:\freqs.pdf'; proc print data=freqs; run; ods pdf close;</pre>                                                                                               | FREQS.PDF contains the standard formatted output from PROC PRINT.                                                                                                |
| Customizing headers and footers          | Specify style attributes and values before the quoted text in a TITLE or FOOTNOTE.<br><b>attribute=style_value</b>                                  | <pre>Options nodate; Title1 j=left font=courier bold 'Frequencies'; Footnote1 height=8pt j=right "&amp;sysdate";</pre>                                                           | Option NODATE removed the standard date and time. The PDF displayed a left justified, courier, bold header, and a date as a small right footer.                  |
| Changing the look of the overall PDF     | Using ODS PDF, specify the style to use<br><b>style=stylename</b>                                                                                   | <pre>ods pdf file='c:\freqs.pdf' style=sasdocprinter; proc print data=freqs; run; ods pdf close;</pre>                                                                           | PDF output contains the PROC PRINT output formatted using the built-in style SASDOCPRINTER.                                                                      |
| Creating a page of custom formatted text | Using ODS specify a character (.i.e. ^) that will be used by ODS to recognize style information in PUT statements.<br><b>escapechar='character'</b> | <pre>ods pdf file='c:\freqs.pdf'; ods escapechar='^'; data _null_; file print; put "^S={font_size=4}" 'Table of Contents'; run; proc print data=freqs; run; ods pdf close;</pre> | A page was created with the title "Table of Contents" prior to a page of PROC PRINT output. Additional PUT statements may be used to create a table of contents  |
| Customizing PDF Bookmarks                | Using ODS to change the default SAS PDF bookmarks.<br><b>proclabel contents=</b>                                                                    | <pre>ods pdf file='c:\freqs.pdf'; ods proclabel 'Report 1'; proc print data=freqs contents='Frequencies'; run; ods pdf close;</pre>                                              | Change the default PDF bookmarks created by PROC PRINT. The label "The Print Procedure" was changed to "Report 1" and "Data Set Freqs" changed to "Frequencies." |

Together the previous ODS techniques may be used to create dramatic changes in the standard SAS output to PDF. Later sections of this paper describe these ODS techniques in more detail and list some essential ODS techniques for PROC REPORT.

## USING ODS TO CREATE AN OUTPUT DATA SET

As shown in Table 1, I created a data set called FREQS using ODS OUTPUT ONEWAYFREQS=FREQS. I determined the name ONEWAYFREQS by using ODS TRACE ON. I preferred a different data structure for FREQS, so I used the macro GETFREQ (included in the last section) to run the FREQ procedure, create the data set FREQS using ODS, and restructure the data set. The macro GETFREQ only requires a library name and data set name as parameters. Specifying a variable list is optional.

```
%let root = C:\projects\PDF\  
libname mylib "&root";  
%getfreq(mylib,mydata,list=);
```

## CREATING A PDF DOCUMENT

Combining many of the techniques shown in Table 1, the following syntax creates a PDF document with a first page having the title table of contents and a second page of listing using SASDOCPRINTER style:

```
Options nodate nocenter;  
ods pdf file="&root.pdfpaper1_2.pdf" style=sasdocprinter;  
ods escapechar='^';  
ods proclabel 'Table of Contents';  
data _null_ ; file print;  
put "^S={font_size=4}" 'Table of Contents';  
run;  
Title1 j=left font=courier bold 'Report 1: Enrollment';  
Footnote1 height=8pt j=right "&sysdate";  
ods proclabel 'Report 1: Enrollment';  
proc print data=freqs noobs label contents='Frequencies';  
var tablename varname varvalue frequency percent;  
run;  
ods pdf close;
```

The ODS techniques I have shown were discussed in more detail in several papers, so I encourage you to review the titles listed in the references and recommended reading sections. The following section highlights some of the issues and papers I found most valuable in learning these techniques.

## SAS STYLES AND CUSTOM STYLES

There are many valuable papers on SAS styles and their creation or modification using the TEMPLATE procedure. Cartier (2002) and Delaney (2003) both showed how the SAS Explorer frame can be used to view the TEMPLATES environment by clicking on the results tab, VIEW menu, and TEMPLATES option. Within the TEMPLATES environment, the built-in SAS styles may be viewed by clicking the SASHELP.TMPLMST folder and the STYLES subfolder. The SASDOCPRINTER style I used in the previous examples is on the list, as well as many others. You may also use the following syntax to print a list of available styles:

```
ods output stats=stylesout;  
proc template;  
  list styles;  
run;  
ods output close;  
proc print data=stylesout; run;
```

While this paper used a built-in SAS style, it is possible to modify templates and create custom styles (Cartier, 2002; Delaney, 2003; Gupta, 2001; McNeil, 2001; Rath, 2002). Delaney (2003) used PROC TEMPLATE to label PDF secondary bookmarks. He pointed out that the PRINT, REPORT, TABULATE, and FREQ for multi-way tables were the only procedures that have the CONTENTS option necessary to rename secondary bookmarks, so he demonstrated a technique for labeling secondary bookmarks using a modified template. McNeill (2001) and Bessler (2005) were also good references for the use of style attributes and values in TITLE, FOOTNOTE, and PUT statements, as well as procedures. If you are interested in understanding how templates, styles and the other techniques in this paper fit into the overall SAS ODS picture, I recommend Gupta (2001).

## ODS AND PROC REPORT

This section covers some techniques to further enhance the ODS output you can obtain by using PROC REPORT. You will not learn how to use the REPORT procedure in this section, but if you do not already know PROC REPORT these powerful ODS techniques will hopefully inspire you to learn. I'll cover two types of techniques: (a) using the STYLE option to PROC REPORT or to a DEFINE statement and (b) using a CALL DEFINE routine within a

COMPUTE block to change a style. Together these two techniques seem to command a high degree of formatting flexibility for a relatively small effort to learn. Both of these techniques build on the same style form used in the previous section: `STYLE = [attribute=value]`.

The first row of Table 2 shows syntax used to change the background color of a report to light blue. In this context the style option requires a parameter that identifies the aspect of the report to change, thus specifying REPORT in parentheses applies the style change to the whole report. Other aspects of the report you can specify include HEADER, COLUMN, LINES, or SUMMARY. The style attributes you specify will vary depending on the aspect of the report you are trying to change. A SAS procedures guide covering PROC REPORT is a good source to determine available style attributes and values (see SAS Institute Inc, 1999). The following website also lists SAS style attributes and values that may be used with SAS procedures TABULATE and REPORT.

<http://v8doc.sas.com/sashtml/proc/z0060410.htm#z0591394>

**TABLE 2. ESSENTIAL ODS TECHNIQUES FOR PROC REPORT**

| Purpose                                                          | Technique                                                                                                                                   | Example Syntax                                                                                                                                           |
|------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| Customize the overall report.                                    | Use PROC REPORT STYLE option and REPORT parameter to change the default style.<br><b>style(report)=[attribute=value]</b>                    | <pre>proc report   style (report)=[background=lightblue];</pre>                                                                                          |
| Customize a line of text to appear before the body of the report | Use PROC REPORT STYLE option and LINES parameter to change the default style.<br><b>style(lines)=[attribute=value]</b>                      | <pre>proc report style (lines)=[font_size=6]; compute before _page_ / left; Line @1 'Action: Check unexpected values shown in red'; endcomp;</pre>       |
| Color a column of data                                           | Use the DEFINE statement STYLE option.<br><b>style =[attribute=value]</b>                                                                   | <pre>define var/ display   style=[background=red];</pre>                                                                                                 |
| Color a row based on the values of the data                      | Use the CALL DEFINE routine in a COMPUTE block to change a default style for a row based on the value of the data.                          | <pre>define t /computed noprint; compute t;   t = order;   if t = 1 then call define(_row_,"style","style={background= cornflowerblue}"); endcomp;</pre> |
| Color a specific cell based on its value                         | Use Proc FORMAT to create a format where a color is specified. Use the DEFINE statement STYLE option to set background equal to the format. | <pre>proc format;   value \$uxv '999'='red'             '988' = 'red'; run; define varvalue / display style=[background=\$uxv.];</pre>                   |

The second row of Table 2 demonstrates the use of STYLE to customize LINE statements throughout the report. In the example, the text “Action: Check unexpected values shown in red” is changed to font size 6 and positioned left justified just above the body of the report. The third row demonstrates STYLE as an option to a DEFINE statement. The DEFINE statement is usually used to identify a variable appearing as a column in the REPORT procedure, thus by specifying `style=[background=red]` the column of values in the report is colored red.

The fourth row in Table 2 shows how ODS and the REPORT procedure allow the style of a row to be set based on the values of a variable in the procedure. In data set FREQS the variable ORDER is equal to 1 at the beginning of each logical block of observations, so the syntax sets the computed variable T equal to ORDER and then tests for the value of T. When T = 1 the current row’s background style is set to the color “cornflowerblue” which is just a bit darker than the blue background for the whole table.

The fifth row of Table 2 shows how a format may be assigned to a STYLE background attribute so that only cells of a certain value are given a certain background color.

The following syntax combines many of the previous techniques and produced the output in Figure 1.

```

%let root = C:\projects\PDF\;
libname rslib "&root";
%let lib = rslib;
options nodate nonumber nocenter orientation=landscape;
footnote1 height=8pt j=right "&sysdate";
Title1 j=center font=courier bold height=8 'Report 1: Frequencies of Student
Enrollment Variables';
ods pdf file="&root.pdfpaper1_2.pdf" style=sasdocprinter;
Proc report data=&lib..freqs nowd style(report)=[background=lightblue just=c]
style(lines)=[font_size=6];
columns dataset varname order varvalue frequency percent t;
define dataset/ group noprint;
define varname /group display;
define order /order display noprint;
define varvalue / display style=[background=$uxv.];
define frequency / display;
define percent / display;
define t /computed noprint;
compute t;
    t = order;
    if t = 1 then call define(_row_,"style","style={background=cornflowerblue}");
endcomp;
compute before _page_ / left;
Line @1 'Action: Check unexpected values shown in red';
endcomp;
run;
ods pdf close;

```

FIGURE 1: PROC REPORT OUTPUT WITH ODS FORMATTING

### Report 1: Frequencies of Student Enrollment Variables

| Action: Check unexpected values shown in red |                |           |         |
|----------------------------------------------|----------------|-----------|---------|
| Variable Name                                | Variable Value | Frequency | Percent |
| P1_GENDER                                    | 1              | 63        | 70.00   |
| P1_GENDER                                    | 2              | 27        | 30.00   |
| P1_LANG                                      | 1              | 67        | 74.44   |
| P1_LANG                                      | 2              | 21        | 23.33   |
| P1_LANG                                      | 988            | 2         | 2.22    |
| P1_LUNCH                                     | 1              | 8         | 8.89    |
| P1_LUNCH                                     | 2              | 50        | 55.56   |
| P1_LUNCH                                     | 3              | 1         | 1.11    |
| P1_LUNCH                                     | 999            | 31        | 34.44   |
| PROJ_ID                                      | 1              | 90        | 100.00  |

## MACRO GETFREQ

```
%macro getfreq(lib,din,list=_all_);
ods listing close;
ods output onewayfreqs=temp;
proc freq data=&din;
  tables &list / missprint;
  format _all_;
run;
ods output close; ods listing;
  proc sort data=temp nodupkey out=list;
    by table; run;
  data _null_;
    set list;
    call symput("v"||trim(left(_n_)),trim(left(scan(table,2))));
    call symput("vc",trim(left(_n_)));
  run;
proc sort data=temp;
  by table; run;
data temp2(keep= dataset varname varvalue frequency percent order);
  set temp;
  by table;
  retain order 0;
  length dataset $30. varname $25. varvalue $35.
  frequency percent order 8. ;
  dataset = "&din";
  label dataset = "Data Set"
  varname="Variable Name" varvalue="Variable Value"
  frequency = "Frequency" percent="Percent" order="Value Order";
  if first.table then order = 1;
  else order=order+1;
  varname = scan(uppercase(table),2);
  %do i = 1 %to &vc;
%let varname=%uppercase(&&v&i);
  %if &i = 1 %then %do;
    if varname =uppercase("&varname") then
      varvalue = trim(left(f_&varname));
  %end;
  %else %do;
    else if varname =uppercase("&varname") then
      varvalue = trim(left(f_&varname));
  %end;
%end;
run;
%let trackd=;
%let trackd = %sysfunc(Exist(&lib..freqs));
%if &trackd > 0 %then %do;
  data &lib..freqs;
    set &lib..freqs temp2; run;
%end;
%else %do;
  data &lib..freqs;
    set temp2; run;
%end;
proc delete data=list temp temp2;run;
%mend;
```

## CONCLUSION

The ODS techniques in this paper included capturing one-way frequency output to a data set, creating a PDF document, changing styles, controlling page setup (e.g. orientation, date, title, and footnote), using PUT, and formatting and traffic-lighting PROC REPORT.

## REFERENCES

- Bessler, L. (2005). Getting started with, and getting the most out of, SAS ODS PDF: No mastery of PROC TEMPLATE required. SAS Conference Proceedings: Technical Solutions, Phuse, Heidelberg, Germany.
- Cartier, J. (2002). Visual styles for V9 SAS output. SAS Conference Proceedings: Technical Solutions, SUGI 27,
- Delaney, K.P. (2003) ODS PDF: It's not just for printing anymore. SAS Conference Proceedings: Data Presentation Section, SUGI 28, Seattle, Washington, US.
- Gupta, S. K. (2001). Using styles and templates to customize SAS ODS output. SAS Conference Proceedings: Advanced Tutorials, SUGI 26, Seattle, Washington, US.
- McNeill, S. (2001) Changes and Enhancements for ODS by Example (through version 8.2). SAS Conference Proceedings: Advanced Tutorials, SUGI 26, Long Beach, California, US.
- Rath, V. (2002). How to change the template and table of contents for SAS web applications. Washington DC, SAS Users Group 2002 First Quarter Meeting (<http://dc-sug.org/pastprog.html>).

## RECOMMENDED READING

SAS Institute Inc. (1999). The complete guide to the SAS Output Delivery System, Version 8. Cary, NC: SAS Institute Inc.

SAS Institute Inc. (1999). SAS Procedures Guide, Version 8. Cary, NC: SAS Institute Inc.

## CONTACT INFORMATION

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

S. Patrick Thornton Ph.D., Sr. Social Science Programmer  
SRI International, Center for Education and Human Services  
Phone: 650 859-5583  
Fax: 650 859-2861  
Email: [patrick.thornton@sri.com](mailto:patrick.thornton@sri.com)  
Web: [www.sri.com](http://www.sri.com)

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.