

Smooth Writing with In-Line Formatting

Louise S. Hadden, Abt Associates Inc., Cambridge, MA

ABSTRACT

PROC TEMPLATE and ODS provide SAS® programmers with the tools to create customized reports and tables that can be used multiple times, for multiple projects. For ad hoc reporting or the exploratory or design phase of creating custom reports and tables, SAS® provides us with an easy option: in-line formatting. There will be a greater range of in-line formatting available in SAS 9.2, but the technique is still useful today in SAS versions 8.2 and 9.1, in a variety of procedures and output destinations. Several examples of in-line formatting will be presented, including formatting titles, footnotes and specific cells, bolding specified lines, shading specified lines, and publishing with a custom font.

INTRODUCTION

In-line formatting within procedural output is generally limited to three procedures: PROC PRINT, PROC REPORT and PROC TABULATE. The exception to this rule is in-line formatting with the use of ODS ESCAPECHAR, which will be briefly discussed in this paper. Even without the use of ODSESCAPECHAR, the number of options with in-line formatting is still exciting and almost overwhelming. As I was preparing to write this paper, I found that I kept thinking of more ways to customize output than would be possible to explore at any given time. The scope of the paper therefore will be limited to a few examples outputting to two destination "families", ODS PRINTER (RTF and PDF) and ODS HTML (HTML4). The examples presented are produced using SAS 9.1.3 (SP4) on a Microsoft XP operating system.

The pros and cons of using in-line formatting versus using customized tagsets and styles will also not be debated in this paper. The paper is mainly intended as a "how to" for in-line formatting, for the general user. Readers are urged to look at the many excellent papers and URLs listed in the "Recommended Reading" sections at the end for more exciting details on in-line formatting within different destinations.

CONSIDERING INHERITANCE AND DOMINANCE

The relationship between tagsets (destinations) and styles is a complex one. Even within SAS-provided tagsets and styles, you can find conflicts. At one point, using the PDF destination and the Minimal style resulted in a lurid green background color, surely not intended by the architects of the style! The complexity increases when adding in-line formatting (or style overrides) to your SAS code. Unexpected results may and often do occur. It is best to use the most basic style designed for the tagset (for example, the RTF style for the RTF tagset) and/or create your own style by modifying a SAS-provided style for certain attributes. The fewer the "frills" in the style you use, the less likely it is to conflict with the instructions given using in-line formatting. Examples using different styles, including a modified style, will be presented.

FONT-WISE AND FANCY-FREE

SAS® provides a number of system fonts to users, including most common fonts such as Arial, Courier, Helvetica, etc. along with the SAS monospace font. However, users are not limited to these fonts. PROC FONTREG will allow you to register other fonts in your SAS installation. Note that you must have administrator rights to register new fonts. In order to specify a custom font in in-line formatting, the font must be available in your system library (usually in your windows\fonts subdirectory on the PC platform) and registered via PROC FONTREG.

```
proc fontreg msglevel=verbose;
    fontpath "%sysget(systemroot)\fonts";
run;
```

This piece of code will attempt to register **all** fonts found in your fonts subdirectory. Here are the abbreviated results in the SAS log:

```
NOTE: The font Klingon (Style: Regular, Weight: Normal) has been added to the SAS
Registry at [CORE\PRINTING\FREETYPE\FONTS\<ttf>Klingon]. Since it is a TRUETYPE font,
it must be referenced as <ttf> Klingon in SAS. The font resides in file
C:\WINDOWS\fonts\Klingon.TTF.
```

STARTING AT THE BEGINNING (AND AT THE END)

Titles and footnotes are obvious targets for in-line formats. The syntax for in-line formatting for **procedural** output will vary by procedure and destination. However, this is not true of in-line formatting for titles and footnotes. In-line formatting of titles and footnotes is also not limited to PROC PRINT, PROC REPORT and PROC TABULATE. However, you may run into conflicts with the style templates for other procedures. Style items that can be specified using in-line formatting for titles, footnotes and ODS text include: font color, bold, italic, font, justification, height, super and subscripts, background color, images such as logos, links to URLs, and even animation (PRINTER destinations only.) In-line formatting of text strings such as titles and footnotes also allows stop and start of different styles within the text string, allowing superscripts, subscripts, color change, justification and font weight change midstream. Several examples are presented below.

At top of program:

```
title1 justify=left color=red height=8pt font=Arial bold "Dropout Analysis for
Oversample Groups" ;
footnote1 color=purple height=8pt font=Arial italic justify=left bold "Program:
kpsamp2.SAS - last run &sysdate";
```

Within ODS statements:

```
title2 justify=center font=arial italic bold color=blue height=8pt "Test Titles -
center justification";
title3 justify=right color=green font=arial height=8pt "{\animtext2 This title is
right-justified and animated}";
```

Results in:

Dropout Analysis for Oversample Groups

Test Titles - center justification

This title is right-justified and animated

....Table

Program: kpsamp2.SAS - last run 05MAR07

The ODS text command within ODS statements in "Printer" destinations such as RTF and PDF allows the user to insert lines of text after titles and procedures at will. There seems to be a more limited ability to format this field using in-line formatting (at least to date I have not experimented enough to figure out what works and what doesn't!) It has something to do with the usertext style element.

```
ods rtf text='{ \animtext1 This is an example of inserting a text field.'};
```

Results in:

This is an example of inserting a text field.

Note that titles inside and outside of the ODS statements may behave differently with respect to in-line formatting, depending on your destination. In general, in-line formatting of titles and footnotes is a little riskier with the HTML family of destinations. The MSOFFICE2K tagset is an example where you may have to use traditional HTML syntax to apply in-line formatting to titles and footnotes, while the HTML4 destination (the default in V9.1) yields some unexpected results.

```
ods html4 file='betterinhtml.xls' path=odsout style=styles.minimal;

proc print data=table1 noobs label uniform split='*' style(header)={font_face='Arial'
font_size=1 font_weight=bold justify=left};
label xlab='Variable Description' xval='Variable Value' totaln='Unweighted*N'
wtotn='Weighted*N' totper='%' chisqp='Chi-Squared*P-Value';
var xlab xval / style={justify=left font_size=1};
var totaln wtotn / style={font_size=1};
var totper / style={font_size=1 htmlstyle= "vnd.ms-excel.numberformat:0.0%"};
```

```

var chisqp / style={font_size=1 foreground=sigf. htmlstyle= "vnd.ms-
excel.numberformat:0.0000"};
title2 color=blue bcolor=green height=2 justify=left "3 way Disability Status"
bcolor=pink height=2 color=black justify=right "Based on ADLCAT";
title3 color=green bcolor=cyan height=2 justify=left
link="http://www.abtassociates.com" "click me!";
run;

ods html4 close;

```

Variable Description	Variable Value	Unweighted N	Weighted N	%	Chi-Squared P-Value
Gender	Male	435	39,014	22.6%	0.0000
Gender	Female	1,595	133,875	77.4%	0.0000
4 Way Age Category	18-34	192	11,291	6.5%	0.0000
4 Way Age Category	35-54	1,246	103,766	60.1%	0.0000
4 Way Age Category	55-64	439	42,124	24.4%	0.0000
4 Way Age Category	65+	150	15,399	8.9%	0.0000
Marital Status	Married	1,349	115,145	66.7%	0.0038
Marital Status	Widowed	89	8,911	5.2%	0.0038
Marital Status	Divorced or Separated	348	30,604	17.7%	0.0038
Marital Status	Never Married	241	17,907	10.4%	0.0038
Race	Caucasian	1,767	150,537	87.1%	0.0236
Race	African-American	101	8,658	5.0%	0.0236
Race	Other	162	13,694	7.9%	0.0236

Looking at the output above, you can see that the effect of splitting formatting in the second title statement is to put the separate text fields into separate columns. This could be useful, but not necessarily expected! The bottom line (or footnote) with regard to in-line formatting of titles, footnotes and ODS text is to experiment and review your output carefully to avoid any stylistic pitfalls.

IN-LINE STYLES IN PROCEDURAL OUTPUT

As briefly mentioned above, in-line formatting is primarily used with three procedures: PROC PRINT, PROC REPORT and PROC TABULATE. Each procedure has a different syntax for applying in-line formats. Some, but not all, of this syntax also varies with the output destination. Along with the PROC PRINT/HTML4/STYLE=MINIMAL example presented above, additional examples using PROC REPORT and PROC TABULATE and the RTF destination will be presented and discussed.

PROC PRINT/HTML4/STYLE=MINIMAL

Using the program above as an example, there are several important items to note. The first is that PROC PRINT allows multiple VAR statements to allow for different formatting in "column" variables, just as PROC REPORT has multiple DEFINES and PROC TABULATE can have multiple CLASS and VAR statements. Style commands follow a forward slash in the VAR statement, enclosed in curly brackets. The syntax is VAR YOURVAR / style={style elements here}. Style elements can be stacked up inside the curly brackets. The default font size in

STYLES.MINIMAL is larger than I wanted, so the first modification I made was to specify font_size=1 in all of my VAR statements. This could also be done by modifying the PROC PRINT statement to include a style(data) statement as well as a style(header) statement. Other items such as font color (using a user-defined format) and justification are specified in various VAR statements. Something that is unique to HTML destinations is the HTMLSTYLE statement. This allows special formatting overrides that Excel recognizes, for example showing leading 0s or decimal places of 1 or 4. The HTMLSTYLE statement itself will vary with the version of SAS and Excel you are using. For my installation, I use the "vnd.ms-excel.numberformat:" version (Excel 97). Others (for example, my co-working running SAS 9.1) may use "mso-number-format:" version. You can modify a style template fairly easily to always allow you to specify an HTMLSTYLE for zip codes and the like so that they "print" correctly, but for a quick and dirty report, the "long way" works fine. This is the cure for unformatted dates, leading zeros, numbers that should be characters, and that pesky scientific notation for numbers larger than 11 digits. HTMLSTYLE is available in all three procedures' style statements when producing output for the HTML family of destinations. You can also modify such items as borders and text rotation through HTMLSTYLE (border, mso-rotate, and so on.) Papers by Chevell Parker and Vince DelGobbo (listed at the end of this paper) are good references for the use of HTMLSTYLE elements.

PROC REPORT/RTF/STYLE=NMEP

PROC REPORT is a more powerful procedure than PROC PRINT and there are many more opportunities for in-line formatting. The ability to define variables within the procedure and to arrange columns in more complex ways allows for additional refinements. One such refinement is to BOLD or SHADE specific lines. In this case, BOLDIT is a variable created in a data step, but it could also be created "on the fly". In addition, this program features the use of superscripting via ODS ESCAPECHAR. A user-written style was used, based on the MINIMAL style supplied by SAS. *(Code to product this style is available from the author via email.)*

```
options ps=55 ls=175 errorabend missing=' ' orientation=landscape nocenter nodate
nonumber;

ods rtf file='MMWR_Table5_test.doc' path=odsout style=styles.nmep bodytitle;
ods escapechar = "^";

proc report nowd data=table5 spacing=8
  style(report)=[cellpadding=3pt vjust=b]
  style(lines)=[just=left font_face=Arial]
  style(header)=[just=center font_face=Arial font_weight=bold font_size=8pt]
  ls=176;
  columns fmtvar ('Main Partner ^{super **} ' ( ('Sex' count1 colper1)
    ('Unprotected Sex^{super****}' count2 colper2) ) )
    ('Non-Main Partner ^{super***}' ( ('Sex' count3 colper3)
    ('Unprotected Sex^{super****}' count4 colper4) ) )
    count boldit;
  define boldit / display ' ' noprint;
  define fmtvar / style(COLUMN)={just=l font_face=Arial
    font_size=8pt cellwidth=300 asis=on}
    style(HEADER)={just=l font_face=Arial font_weight=bold
    font_size=8pt cellwidth=300 asis=on};
  define count1 / style(COLUMN)={just=c font_face=Arial
    font_size=8pt cellwidth=100 asis=on}
    style(HEADER)={just=c font_face=Arial font_weight=bold
    font_size=8pt cellwidth=100 asis=on};
  . . .
compute boldit;
  if (boldit eq 1) then call define(_row_,"STYLE","STYLE=[FONT_WEIGHT=BOLD]");
endcomp;

title1 justify=left font="Arial" height=8pt color=blue bold "Table 5. Number^{super*}
and percentage of participants reporting having had";
title2 justify=left font="Arial" height=8pt color=blue bold 'Sex with a main or casual
partner during the preceding 12 months, by selected characteristics';
footnote1 justify=left font="Arial" height=8pt color=red italic "^{super*}Numbers
might not add up to total because of missing data";
```

```

footnote2 justify=left font="Arial" height=8pt italic "^{super**}A man with whom the
participant had sex with and to whom he felt most committed (e.g., boyfriend, spouse,
significant other, or life partner)";
footnote3 justify=left font="Arial" height=8pt italic "^{super***}A man with whom the
participant had sex with but who was not considered a main partner";
footnote4 justify=left font="Arial" height=8pt italic "^{super****}Neither the
participant nor his partner used a condom. Proportion reported is that of all
participants who engaged in Sex with that type of partner.";
run;

ods rtf close;
quit;

```

MMWR_Table5_test.doc - Microsoft Word

File Edit View Insert Format Tools Table Window Help PDF Create! Type a question for help

Normal + 12 pt Times New Roman 12 B I

I love Tagsets! Make your Reports Stand Out!

Table 5. Number and percentage of participants reporting having had Sex with a main or casual partner during the preceding 12 months, by selected characteristics

Click for more information

Characteristic	Main Partner ^{**}				Non-Main Partner ^{***}			
	Sex		Unprotected Sex ^{***}		Sex		Unprotected Sex ^{***}	
	No.	(%)	No.	(%)	No.	(%)	No.	(%)
Age Group in Yrs								
18 - 24	69	(69)	47	(68)	48	(48)	44	(92)
25 - 34	138	(60)	90	(65)	115	(50)	107	(93)
35 - 44	171	(54)	127	(74)	166	(52)	158	(95)
45 - 54	36	(31)	22	(61)	53	(45)	51	(96)
ge 55	11	(24)	9	(82)	27	(60)	25	(93)
Boston, Massachusetts	425	(52)	295	(69)	409	(50)	385	(94)

^{**} Numbers might not add up to total because of missing data

^{***} A man with whom the participant had sex with and to whom he felt most committed (e.g., boyfriend, spouse, significant other, or life partner)

Page 1 Sec 1 1/1 At 3.5" Ln 16 Col 1 REC TRK EXT OVR

Note that the "LINK=" in the title statement operates as it does for the HTML4 example above but behaves somewhat differently in the document. When the user hovers over the title there are instructions to type CNTL and click on the title to get to a website instead of simply clicking. The superscript command is enclosed in brackets following the ODS ESCAPECHAR. The subscript command operates in a similar fashion but is not demonstrated here. The "boldit" field is set as a non-printing variable in the DEFINE and COMPUTE statements, but must be included in the COLUMNS statement. To shade specified lines the same is true (see output below.)

TableB7_shaded.doc - Microsoft Word

File Edit View Insert Format Tools Table Window Help PDF Create! Type a question for help

Normal + Arial, Arial 10 B I

Table B7. Knowledge Questions: Proportion who knew
by Knowledge of www.medicare.gov and 1-800-Medicare

Knowledge Question	Aware of Website			Aware of Phone Number		
	N for Website Knowledge	Knows of Medicare website	Does Not Know of Medicare website	N for Toll-Free Number Knowledge	Knows of Toll-Free Number	Does Not Know of Toll-Free Number
General Medicare Program Knowledge						
Medicare does not pay all health care expenses	204	99%	88%	205	99%	84%
Part A covers hospital costs and Part B	196	96%	84%	198	97%	78%

Page 1 Sec 1 1/2 At 0.6" Ln 1 Col 1 REC TRK EXT OVR

PROC TABULATE/RTF/STYLE=MINIMAL

PROC TABULATE, like PROC REPORT, offers many opportunities for in-line formatting. This program features the use of a different font and color coding specific class variables and their headers using multiple CLASS and CLASSLEV statements. Table cells or the “crosses” and the VAR variables can also be specifically formatted, via the TABLES statement. In addition, the “box” can be formatted. This program was used in conjunction with a post-processing routine to rotate the column headers, available from the author via email. Note that column rotation can be accomplished within style statements for the HTML family of destinations, using MSO-ROTATE commands, but so far, is not available for the RTF destination.

```
filename rtf 'test5b.rtf';

ods rtf file=rtf path=odsout style=styles.minimal bodytitle;

proc tabulate data=anal missing format=20.
  style={bordercolor=black borderwidth=1 cellwidth=75
    font=("Times New Roman",7pt,Normal)};
  var counter mjr_idea mjr_tit1 mjr_cbgr mjr_tit4 mjr_safe mjr_spd
      mjr_sgen mjr_ttax mjr_lott mjr_locl
      mjr_mcd mjr_3rdp mjr_schi mjr_self mjr_foun mjr_othr /
  s={bordercolor=black borderwidth=2 cellwidth=75
    font=("Times New Roman",7pt,Bold)};
  weight basewgt3;
  class region distsz3 / s={background=lightblue bordercolor=black borderwidth=2
    font=("Times New Roman",8pt,Bold)};
  classlev region distsz3 / s={background=lightblue bordercolor=black borderwidth=2
    font=("Times New Roman",8pt)};
  class loc_cat / s={bordercolor=black borderwidth=2
    font=("Times New Roman",8pt,Bold)};
  classlev loc_cat / s={bordercolor=black borderwidth=2
    font=("Times New Roman",8pt)};
  tables all={label='TOTAL' s={font=("Times New Roman",8pt,Bold)
    foreground=white background=blue}}
    region loc_cat distsz3,
  counter*(sum=' '*f=comma9.)
  mjr_idea*(pctsum<counter>=' '*f=pctpic.)
  ...
```

```

/ box={s={bordercolor=black borderwidth=2 background=lightblue }};
title1 font="Times New Roman 9pt" Bold 'Table 5b-Weighted';
title2 font="Times New Roman 9pt" Bold 'Percentage of School Districts For Which the
Source is Among Their Top Five Sources,';
title3 font="Times New Roman 9pt" Bold 'By Selected District Characteristics: 2002-
2003';
footnotel font="Times New Roman 8pt" 'Source: Survey of the Characteristics and
Funding of School Mental Health Services:';
footnote2 font="Times New Roman 8pt" '2002-2003, Center for Mental Health Services,
U.S. Department of Health and Human Services.';
footnote3 font="Times New Roman 8pt" 'District Questionnaire, Item 8';
run;

ods rtf close;
```

test5b.rtf - Microsoft Word

	# of Districts	Q8: IDEA Source of MH Svc Funding	Q8: Title I Source of MH Svc Funding	Q8: Comm MH Svc Block Grant Source of MH Svc Funding	Q8: Title 4 Source of MH Svc Funding	Q8: Safe Schools/Healthy Student Int Svc of MH Svc Funding	Q8: State Special Education Funds Source of MH Svc Funding	Q8: State General Fund Source of MH Svc Funding	Q8: Tobacco Tax/Settlement Source of MH Svc Funding	Q8: Lottery Funds Source of MH Svc Funding	Q8: Local Funds (Taxes) Source of MH Svc Funding	Q8: Medicaid Reimbursement Source of MH Svc Funding	Q8: 3rd Party Payments Source of MH Svc Funding
TOTAL	1,533	50.9%	15.0%	7.9%	54.4%	17.1%	36.8%	23.3%	13.3%	0.6%	19.0%	27.6%	3.9%
Region													
Midwest	270	57.2%	8.8%	9.8%	35.5%	18.8%	36.7%	37.0%	15.4%	0.0%	29.2%	48.6%	0.0%
West	1,262	49.5%	16.4%	7.5%	58.5%	16.8%	36.8%	20.3%	12.8%	0.7%	16.9%	23.1%	4.8%
Urbanicity													
Urban	557	44.2%	8.0%	4.3%	48.4%	9.9%	36.5%	14.9%	24.0%	0.0%	26.1%	11.9%	7.9%
Suburban	434	51.4%	8.2%	18.5%	68.2%	23.3%	32.7%	32.3%	5.4%	2.3%	6.4%	23.8%	3.1%
Rural	542	57.2%	27.8%	3.1%	49.6%	19.7%	40.4%	24.6%	8.6%	0.0%	21.9%	46.7%	0.6%
District Size													
1-5 Schools	1,148	42.7%	13.9%	3.8%	51.4%	12.1%	31.4%	16.6%	12.9%	0.0%	20.6%	25.0%	3.8%
6-15 Schools	290	74.5%	13.1%	18.9%	64.1%	31.7%	55.6%	43.6%	13.7%	3.4%	12.3%	32.0%	3.4%
16+ Schools	95	76.5%	34.8%	23.6%	61.7%	33.2%	45.4%	41.2%	16.3%	0.0%	21.4%	45.5%	7.1%

PROC REPORT/PDF/STYLE=NMEP/FONT=KLINGON!

I have **always** wanted to publish in Klingon. Once I had attended Scott Huntley’s presentation at SUGI 31 (“Let the ODS PRINTER Statement Take Your Output into the Twenty-First Century”) I knew I had found the answer. A little surfing on the internet yielded a gold mine of Star Trek fonts. PROC FONTREG allowed me to register the fonts, and then the Klingon font in all its glory was available to the SAS system. Just as Arial or Times Roman can be specified for a font face, so can Klingon. The downside? If you email your wonderful Klingon documents to users without these fonts on their systems, the non-standard fonts will not appear in documents produced for destinations that do not embed fonts (at this point, only the PDF destination embeds fonts.) Klingon will be replaced by a far more prosaic font.

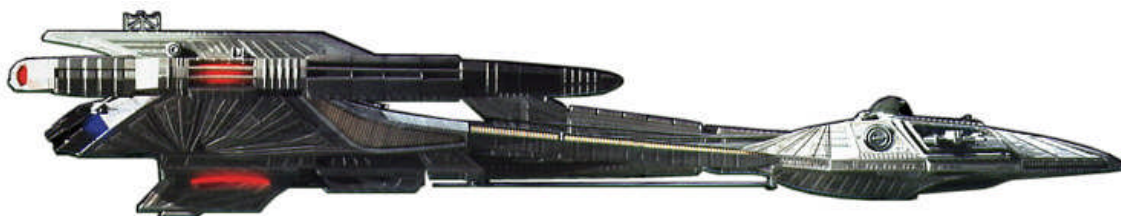
```
ods pdf file='TableB7_in_klingon.pdf' style=styles.nmep;

proc report nowd data=printb2 spacing=8
  style(report)=[cellpadding=3pt vjust=b]
  style(lines)=[just=left font_face=Klingon]
  style(header)=[just=center font_face=Klingon font_weight=bold font_size=8pt]
  ls=176;
  columns linetit (('Aware of Website' website_total pctwebsitel pctwebsite2)
    ('Aware of Phone Number' phone_total pctphonel pctphone2)) boldit;
  define boldit / display ' ' noprint;
  define linetit / style(COLUMN)={just=l font_face=Klingon
    font_size=8pt cellwidth=350 asis=on}
    style(HEADER)={just=l font_face=Klingon font_weight=bold
    font_size=8pt cellwidth=350 asis=on};
  define pctwebsitel / style(COLUMN)={just=c font_face=Klingon
    font_size=8pt cellwidth=90 asis=on}
    style(HEADER)={just=c font_face=Klingon font_weight=bold
    font_size=8pt cellwidth=90 asis=on};
  . . .

  define phone_total / style(COLUMN)={just=c font_face=Klingon
    font_size=8pt cellwidth=105 asis=on}
    style(HEADER)={just=c font_face=Klingon font_weight=bold
    font_size=8pt cellwidth=105 asis=on};
  compute boldit;
  if (boldit eq 1) then call define(_row_,"STYLE","STYLE=[FONT_WEIGHT=BOLD]");
  endcomp;

  title1 justify=left font=Klingon bold "Table B7. Knowledge Questions: Proportion who
  knew";
  title2 justify=left font=Klingon bold 'by Knowledge of www.medicare.gov and 1-800-
  Medicare';
  footnote1 justify=left font=Klingon bold italic 'Notes: Awareness of Mammogram
  Coverage Limited to Women; Awareness of Prostate Exam Coverage Limited to Men.';
  footnote2 justify=left font=Klingon bold italic "Source: New Beneficiary Survey for
  &birthmo &birthyr Birthdays
  - Round &round";
run;

ods pdf close;
quit;
```





SAS® provides us with many tools to customize ODS output. In-line formatting provides SAS® programmers with an easy option for finding formatting that works on an experimental basis, and incorporating these formats and styles into custom tagsets and styles for repeated use. Ka'pla.

SAS® Online Documentation PC SAS V9.1

Haworth, Lauren. 1999. *PROC TABULATE By Example*. Cary, NC: SAS Institute Inc.

SUPPORT.SAS.COM – the samples, FAQs and human beings behind the scene are the greatest!

Scott Huntley for his inspirational talk at SUGI 31 on the future of reporting in 9.2.

Eric Gebhart, Vince DelGobbo and Chevell Parker for introducing me to in-line styles (or style overrides as Vince prefers) in HTML-based tagsets.



Sandy McNeill, Lauren Haworth and Ray Pass for introducing me to the use of in-line styles in PROC TABULATE and PROC REPORT.

Cynthia Zender for helping diminish my fears of PROC TEMPLATE, and her worthy paper and presentation on ODS ESCAPECHAR.

Wayne Hester for his generous help with my problems with the RTF destination, and for coming up with really cool additions to the RTF toolset.

Leroy Bessler for his eye-popping, ear-splitting artistry in ODS.

I couldn't have done it without them! Please read their papers listed in "Recommended Reading" below, and/or attend any of their presentations.

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.

RECOMMENDED READING

Bessler, Leroy. "Multi-Media Wizardry: How to Make ODS Outputs That "Dance and Sing."" Proceedings of the Thirty-First Annual SAS® Users Group International Conference. March 2006.

Brown, David. "%sas2xl: A Flexible SAS® Macro That Uses Tagsets to Produce Complex, Multi-Tab Excel Spreadsheets with Custom Formatting." Proceedings of the Thirtieth Annual SAS® Users Group International Conference. April 2005.

DelGobbo, Vince. "A Beginner's Guide to Incorporating SAS® Output in Microsoft® Office Applications." Proceedings of the Twenty-Eighth Annual SAS® Users Group International Conference. April 2003.

DelGobbo, Vince. "Creating *AND* Importing Multi-Sheet Excel Workbooks the Easy Way with SAS®." Proceedings of the Thirty-First Annual SAS® Users Group International Conference. March 2006.

Fan, Zizhong. "Make the Invisible Visible – A Case Study of Using ODS Inline Formatting Style." Proceedings of the Thirtieth Annual SAS® Users Group International Conference. April 2005.

Feder, Steven. "Perfecting Report Output to RTF." Proceedings of the Seventeenth Northeast SAS® Users Group Conference, November 2004.

Gebhart, Eric. "The Beginners Guide to ODS MARKUP: Don't Panic." Proceedings of the Thirty-First Annual SAS® Users Group International Conference. March 2006.

Haworth, Lauren. "SAS® with Style: Creating your own ODS Style Template." Proceedings of the Twenty-Eighth Annual SAS® Users Group International Conference. April 2003.

Hester, Wayne. "Teaching your RTF Tagset to Do Clever Tricks." Proceedings of the Thirty-First Annual SAS® Users Group International Conference. March 2006.

Huntley, Scott. "Let the ODS Printer Statement Take Your Output into the Twenty-First Century." Proceedings of the Thirty-First Annual SAS® Users Group International Conference. March 2006.

Mitchell, Rick. "Stranded on a Deserted Island With Nothing But TITLE Statements." Proceedings of the Thirtieth Annual SAS® Users Group International Conference. April 2005.

Parker, Chevell. "Generating Custom Excel Spreadsheets using ODS." Proceedings of the Twenty-Eighth Annual SAS® Users Group International Conference. April 2003.

Pass, Ray and McNeill, Sandy. "PROC REPORT: Doin' It in Style!" Proceedings of the Twenty-Eighth Annual SAS® Users Group International Conference. April 2003.

Pass, Ray and McNeill, Sandy. "PROC TABULATE: Doin' It in Style!" Proceedings of the Twenty-Ninth Annual SAS® Users Group International Conference. May 2004.

Sanjaya, Kumar. "ODS: A Unique Method to Present Data." Proceedings of the Seventeenth Northeast SAS® Users Group Conference, November 2004.

Tong, Cindy. "ODS RTF: Practical Tips." Proceedings of the Sixteenth Northeast SAS® Users Group Conference. September 2003.

Zender, Cynthia. "Funny ^Stuff~ in My Code: Using ODS ESCAPECHAR." Proceedings of SAS® Global Forum 2007, April 2007.

CONTACT INFORMATION

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

Louise Hadden
Abt Associates Inc.
55 Wheeler St.
Cambridge, MA 02138
(617) 349-2385 (work)
(617) 349-2675 (fax)
Email: louise_hadden@abtassoc.com



Sample code is available from the author upon request.



KEYWORDS

SAS®; PROC FONTREG; PROC TABULATE; PROC REPORT; PROC PRINT; TAGSET;
STYLE; PROC FORMAT; TITLE; FOOTNOTE; FONT; KLINGON