

Paper 091-30

PROC TABULATE and ODS RTF:**The Perfect Fit for Complex Tables****Louise Hadden, Abt Associates Inc., Cambridge, MA****ABSTRACT**

With the great advances in output design SAS and ODS offers the SAS programmer, the bar has definitely been raised in terms of producing presentation quality SAS output. Analysts frequently design table formats in packages such as Microsoft Word and Excel, and request that the SAS programmer fit SAS output to their specifications. This paper will present a challenge and solution, using PROC TABULATE and the RTF destination to fit SAS output into a pre-designed template. As the title indicates, a technique to rotate column headers without having to post-process in Microsoft Word will be demonstrated, using PC SAS Version 8.2 and PC SAS Version 9.1, as well as UNIX SAS Version 8.2. The code to produce a sample table will be dissected below, with relevant items highlighted in gray “tip” boxes. The full program and sample table is also available as an appendix.

INTRODUCTION

As with many innovative programming techniques, the subject of this paper arose out of the programmer’s desire to produce the best result with the least amount of in-person intervention possible. I was presented with a series of spreadsheets created by hand in Microsoft Excel, and asked to “fill them in” with data from a national survey.

Previous programmers working on similar projects had used SAS to produce the desired statistics, but cut and pasted both the SAS tabulations and the Microsoft Excel “shells” into Microsoft Word. My challenge became to avoid the pitfalls of irreproducible results (not to mention all that mouse work!), and output SAS tabulations directly into Microsoft Word via the RTF destination or tagset in exactly the format shown in the pre-designed Microsoft Excel templates (sample at right).

DECISIONS, DECISIONS

Since the final reports were to be submitted in Microsoft Word, the choice of an output destination or tagset was easy. RTF, or Rich Text Format, is a markup language easily assimilated into a Microsoft Word document. For a comprehensive subset of possible permutations in text formatting in a Word document, there is a RTF command driving the format. RTF also offers what I think of as “page style” formatting, similar to PDF or any other “printer” destination. This allows for a more controlled formatting process compared to tagsets designed to be viewed on a computer screen or imported into spreadsheet software, where the destination can override desired formats.

	A	B	C	D	E
1	Table 5b.				
2	Percentage of school districts for which th				
3	2002-2003				
4		Individuals with Disabilities Education Act (IDEA)	Title I	Community Mental Health Services Block Grant	Title IV - Safe & Drug-Free Schools and Communities
6	All Districts				
8	Region				
9	Northeast				

The choice of a procedure is much more difficult. There are three primary procedures which allow you to control styles within the procedure via style commands; PROC PRINT, PROC REPORT and PROC TABULATE.

Our old familiar friend and long-time SAS workhorse PROC PRINT cannot hold a candle to PROC REPORT and PROC TABULATE in terms of the ability to control every aspect of SAS output. After attending many of the highly entertaining and informative debates on the two “titans” REPORT and TABULATE sponsored by Pass, Bruns, McNeill et al, I now wholeheartedly believe, with very few exceptions, that if it can be done in REPORT, it can be done in TABULATE, and vice versa. Both procedures allow for computing “on the fly,” presentation of numerous statistics, and wonderful flexibility in exactly how those statistics are presented. Note, however, that the syntax for using procedure level styles differ by both procedure and output destination.

PROC TABULATE

```
proc tabulate data=anal missing format=20.
  style={bordercolor=black borderwidth=1
  cellwidth=118
  font=("Times New Roman",8pt,Normal)};
  var counter schc_rat ... /
  s={bordercolor=black borderwidth=2
  cellwidth=118
  font=("Times New Roman",8pt,Bold)};
  weight int_wgt;
  class region srvlevel ... /
  s={bordercolor=black borderwidth=2
  font=("Times New Roman",8pt,Bold)};
  tables all region srvlevel ...,
  schc_rat*(mean=' '*f=comma9.4)
  / box={s={bordercolor=black
  borderwidth=2}};
  keyword all / s={font=("Times New
  Roman",8pt,Bold)};
  keylabel all='TOTAL';
run;
```

PROC REPORT

```
proc report data=anal nowd split='*'
  spacing=10 style(report)=[cellpadding=12
  vjust=b] style(lines)=[just=left
  font_face=Verdana] ls=150;
  column ...;
  define a / ' ' computed
  style(COLUMN)={just=left
  font_face=Verdana font_size=9pt
  cellwidth=3.75in asis=on
  pretext=" " posttext=" "};
  compute before _page_/left;
  line "\S={font_face=Verdana
  font_size=14pt font_weight=bold}
  Your Title Here";
  endcomp;
run;

*program snippet courtesy of Carolyn Robin-
son, Abt Associates Inc.
```

PROC PRINT

```
proc print data=printit label uniform;
  var var2 chisq;
  var chisqp / style={foreground=temp.};
  var chisqdf;
  id var1;
run;
```

I've been using SAS for over 26 years, and PROC REPORT, however wonderful, just wasn't around in the SAS equivalent of the Iron Age. PROC TABULATE, a much earlier addition to the SAS programmer's repertoire, had (and has) that comforting look and feel of FORTRAN code. PROC TABULATE has come a long way in terms of its capacity and design, and has definitely changed over the years, but remains most secure to me relative to PROC REPORT. My decision from the outset was to use PROC

TABULATE with the RTF destination to produce a complex set of tables.

Familiarity was not the only reason for choosing PROC TABULATE. One of PROC TABULATE's handy attributes is the ability to specify complex denominators “on the fly” and calculate percents based on the specified denominators. This particular set of tables was for a national survey of schools with a complex sample design. The client wished to have weighted numbers presented in a total column alongside columns with weighted percents based on different denominators by column, making PROC TABULATE the perfect fit. For some tables, a combination of Ns, percents and other statistics such as means were desired on a column by column basis, which was also possible in PROC TABULATE.

DOWN TO BUSINESS



Many of the pre-formatted tables I was asked to fill in had identical “row” variables, a set of seven descriptive variables with three or four values. This made it easy to set up a standard set of “class” variables in PROC TABULATE with a standard “style”. Note that you can also have multiple class and classvar statements if you wished to format specific class variables in different ways. For example you might want to shade alternate class variables by adding a background option to your style statement. In this case since the reports were to be reproduced in black and white and grayscale and/or colors can sometimes not reproduce well, the client opted not to differentiate in this way.

The “column” variables differed from table to table. The client wished to have sample numbers or means represented in the first column, and percentages using specific denominators represented in subsequent columns, which can be accomplished with ease with PROC TABULATE. Note that any denominator used must be in the VAR statement even if it is not “printed.”

Note also the use of “cellwidth” in the style option statement. This determines the width of the column in MIPS. It can also be used in the PROC TABULATE statement itself. If present in both the PROC TABULATE and VAR statements, the cellwidth should match.

Cellwidth in both PROC TABULATE and PROC REPORT can assist with fitting in more columns across a page and is a more effective tool than using a format for variables to specify width.

PROC TABULATE generates an automatic heading for statistics unless the programmer specifically blocks it in the table statement. The client did not wish the statistic heading to appear, yet also wished specific statistics to be appropriately identified, for example, with a percent sign following. While this could ordinarily be done with a format, percents that PROC TABULATE calculates “on the fly” are in whole numbers. A special user-defined picture format must be used in this case.

I used the format suggested by Lauren Haworth in “PROC TABULATE By Example,” a wonderful reference tool. Since the client wished the percent to be expressed with one decimal place, I modified the format with 9’s instead of 0’s (thanks to the suggestion of the Code Clinic staff at WUSS 2003). Without the modification, some percents were expressed incorrectly without a decimal place.

The client also wished to experiment using a corporate logo in the tables. PROC TABULATE and PROC REPORT both allow the use of “preimage” graphics files. In PROC TABULATE, a logo can be easily inserted in the “box” in the top left corner of the output. The type of graphics file used is dependent on the SAS output destination: for the RTF destination and PROC TABULATE I used a JPEG file.

TABULATE TIP:

Think INSIDE the box

```
tables all region srvlevel ...,
  schc_ns*(mean*f=comma9.4)
  schc_any*(pctsum<counter>=' '*f=pctpic.)
...
/ box={s={bordercolor=black borderwidth=2
  preimage='abtlogo.jpg'}};
```

A clipped snapshot of the RTF file is to the right.

TABULATE TIP:

Named Denominator

```
tables all region srvlevel ...,
  schc_ns*(mean*f=comma9.4)
  schc_any*(pctsum<counter>=' '*f=pctpic.)
```

TABULATE TIP:

Cellwidth=

```
var counter schc_any ... /
s={bordercolor=black borderwidth=2
  cellwidth=100
  font=("Times New Roman",8pt,Bold)};
```

TABULATE TIP:


Calculated Percent Format

```
proc format;
  picture pctpic low-high='009.9%';
run;
```

To use this format and to eliminate the PCTSUM heading:

```
tables all region srvlevel ...,
  schc_any*(pctsum<counter>=' '*f=pctpic.)
...
```

Table 13A.
Type of School Staff Providing Mental Health Serv
by Selected School Characteristics: 2002-2003

	School Counselors	Any Cot
	Mean	
TOTAL	1.8532	
Region		
Northeast	2.2455	
South	1.6451	
---	

All the abilities of PROC TABULATE and ODS RTF combined to allow me to produce most of the tables fairly easily. Some additional tips I gleaned along the way:

RTF TIP:

When is a .DOC not a .DOC?

RTF output, while it can be saved as a .DOC in the ODS RTF FILE= statement and opened directly in Microsoft Word®, remains an RTF file unless you “SAVE AS” regardless of the file extension. The same is true of HTML and MSOFFICE2K output files which can be saved as .XLS in the ODS xxx FILE= statement. You must SAVE AS to convert these output files from HTML. This is useful to know because both RTF and HTML files take up more space than their MS Word® or Excel® counterparts.

STYLE TIP:

Hierarchy of Style Attributes

When using procedure level styles, care must be taken to account for the hierarchy of style attributes. For example, using a pre-written style in the ODS FILE= statement may conflict and/or be overwritten by a procedure level style, with some unexpected results. In addition, (I discovered the hard way), using procedure level styles on the procedure and subsequent statement lines can cause problems if they are different, such as different values for cellwidth. Consistency is key.

ODS PRINTER TIP:

To rotate your output so that it prints sideways for printer destinations such as RTF and PDF:

Options Orientation=Landscape;

TABULATE TIP:

Effect of Missing Values in Class Variables, or, Where DID Those Observations Go?

These tables were created using data from a survey with a complex sample design, which involved stratification and weighting. There were usually multiple class variables involved, some of which had missing values. If there are any missing values in ANY class variables, PROC TABULATE eliminates those records from the tabulation. This may result in a much lower N than expected for the entire table. Care should be taken to expand the possible values of class variables to include missing if necessary.

UP WORDS AND ROTATE

At some point it had to happen. I reached a table (the template shown above) that I simply could not fit onto a page, despite my best efforts. The column headers as specified by the client were simply too long, and there were too many columns even with short column headers. I knew from experimenting with the too wide output file that I could post-edit in Microsoft Word® to rotate the column headers and fit the table on a page. But, I didn't WANT to have to post-edit. The answer to my dilemma came in my review of Cindy Tong's helpful NESUG '03 paper, "ODS RTF: Practical Tips" and the greatly appreciated technical assistance of Wayne Hester (SAS® R and D.)



RTF, like HTML, is a markup language which contains formatting commands. An RTF file can be viewed in a text editor as a text file. An example is provided to the right.

```
{\rtf1\ansi\ansicpg1252\uc1\deff0\def
{\fonttbl
{\f1\fnil\fpqrq0\fcharset0;}
{\f2\froman\fpqrq2\fcharset0 Times New
}{\colortbl;
\red0\green0\blue0;
\red0\green0\blue255;
\red0\green255\blue255;
\red0\green255\blue0;
\red255\green0\blue255;
\red255\green0\blue0;
\red255\green255\blue0;
\red255\green255\blue255;
\red0\green0\blue128;
\red0\green128\blue128;
\red0\green128\blue0;
\red128\green0\blue128;
\red128\green0\blue0;
\red128\green128\blue0;
\red128\green128\blue128;
\red192\green192\blue192;
}{\stylesheet{\widctlpar\adjustright\
}{\info{\title U8 SAS System Output}}
```


It follows, then, that one could determine the command used to rotate column headers by reviewing the original RTF file produced by SAS and the post-edited file in Microsoft Word®. Looking at these massive files quickly dissuaded me, and I resorted to RT*M, "Rich Text Format (RTF) Specification, Version 1.6" of the Microsoft® Corporation referred to by Cindy Tong. Eventually I found the two commands of interest. Then the challenge became to replace the command for horizontal column headers with vertical column headers without resorting to post-editing in Word®.

After attempting to accomplish my goal unsuccessfully, I enlisted the help of SAS Technical Support. I was referred to Wayne Hester of SAS® R and D and he had the answer with a neat little macro. It did involve post-processing, but, it was post-processing in SAS®, within my reporting program. That I could live with!

Instead of outputting directly to the RTF destination, the SAS output is directed to an intermediate RTF file which is then post-processed in a macro to change the RTF command for column header orientation before being output into the final RTF file. The macro, while used to change the column orientation in my case, could also be used to change other specifications.

The result is a perfectly sized landscape report with rotated column headers, seen below.

RTF CONVERSION MACRO:

```
%macro do_one(source=xxx);
data temp;
length line $400;
infile &source length=lg lrecl=1000 end=eof;
input @1 line $varying400. lg;
data _null_;
set temp; retain mod 0;
file &source;
/* mods are done for the trow header data
only */
if (index(line,"\trowd\tkeep") > 0) then do;
if (index(line,"\trowd\tkeep\t rhdr") > 0)
then mod = 1;
else mod = 0;
end;
if (mod) then do;
/* this works because the commands are the
same width */
line=tranwrd(line,"\cltxlrb", "\cltxblr");
line=tranwrd(line,"\clvertalb", "\clvertalt");
line=tranwrd(line,"\qc", "\ql");
put line;
/* The header cell needs to be tall enough to
accept the change. */
/* You may need to adjust this depending on
what your column headers are. */
/* Measurement is in twips. 20 twips to the
point, 72 points to the inch.*/
if ( index(line,"\trowd\tkeep\t rhdr") > 0)
then put "\trrh-1625";
end;
else put line;
run;
%mend;

%do_one(source=rtf);

*indents and spaces removed; full code sample
at end.
```

	# 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 Funding	Q8: Title 4 Source of MH Svc Funding
TOTAL	14,752	61.5%	22.3%	6.9%	52.8%
Region					
Northeast	3,110	77.0%	17.0%	3.7%	58.2%
South	5,441	54.0%	26.1%	5.6%	53.2%
Midwest	3,330	61.3%	22.1%	7.2%	47.9%
West	2,872	59.4%	21.0%	12.1%	52.2%
Urbanicity					
Urban	2,002	64.0%	25.1%	6.1%	47.5%

STOP! WAIT! Go! See WHAT TRAFFIC LIGHTING CAN DO FOR YOU

ODS in conjunction with PROC REPORT, PROC TABULATE and PROC PRINT also allow you to use background and foreground color in your output to highlight or differentiate columns, rows and variables. In PROC TABULATE, background and foreground colors can be applied in the various style statements (for example, the PROC statement, the CLASS statement(s), the CLASSLEV statements(s), the TABLES statement(s), and the BOX statement.) They can also be applied in a user-defined style template. Colors can be assigned directly in the style

statement, or with a user-defined format. Due to space constraints, I will not demonstrate the highlighting of rows and columns here. Please contact the author for complete code samples.

Example of Traffic-Lighting using a User-Defined Format and PROC PRINT

```

TRAFFIC LIGHTING USING PROC PRINT

proc print data=printit label uniform;
  var var2 chisq;
  var chisq / style={foreground=temp.};
  var chisqdf;
  id var1;

run;
    
```

tab	Chi Square Statistic	Chi Square P-Value Statistic	Chi Square Degrees of Freedom
Services	10.73	0.0979	6
Services	18.50	0.0053	6
Services	0.79	0.9400	4
Services	4.23	0.3757	4
Services	1.29	0.8632	4
Services	2.76	0.5994	4
Services	9.49	0.1490	6

An additional request of the client was to perform significance tests on the tables created for this project using PROC TABULATE. Due to the complex survey sample design, these tables were created in SAS-callable SUDAAN. P-values were output from the PROC CROSSTAB procedure for every tabulation within each PROC TABULATE table, and the resulting data table was output to ODS RTF using PROC PRINT and a user-defined color format to denote significance at 2 levels (5% and 10%).

OH NO! THEY BROKE IT!

A few months after I completed the programming for this project in SAS® Version 8.2, I had SAS® Version 9.1 installed. I wondered how my tables would look running 9.1, and tried one of my programs out. My hope was that the tables would look even better. Alas, my hope was unfounded.

Example of Table 21 produced using SAS® 8.2

Example of Table 21 produced using SAS® 9.1 (no changes to code) – Yikes!

Table 21-Unweighted
Percentage of Schools Collecting Various Types of Data and Types of MH Problems Presented by Selected School Characteristics: 2002-2003

	Has Access to Data	Types of MH Problems Presented by Students	Types of MH Problems Presented by Students
TOTAL	649	66.2%	
Region			
Northeast	137	63.5%	
South	150	67.3%	
Midwest	228	67.9%	
West	134	64.9%	
Level			

Table 21-Unweighted
Percentage of Schools Collecting Various Types of Data and Types of MH Problems Presented by Selected School Characteristics:

	Has Access to Data	Has Access to Data	Types of MH Problems Presented by Students
TOTAL	649	100.0%	
Region			
Northeast	137	100.0%	
South	150	100.0%	

What on earth happened? I used the same style template and the same code as in 8.2, but there's gobbledygook in my box, the titles and footnotes are huge and in the wrong font, and my class variables are huge and in the wrong font.

I turned to the ever helpful folks at SAS® Technical Support, and Bari Lawhorn gave me this helpful tip:

Okay, one problem solved and I didn't have to touch my code. Note that the gobbledegook actually allows you to create a table of contents in your RTF document simply by adding "CONTENTS" at the end of your ODS RTF statement, an experimental feature of 9.1.

RTF files created by SAS now have the ability to utilize a table of contents, so the raw RTF file shows \tc instructions. You should be able to eliminate them from view from within MS Word by:

tools..options..view..
under FORMATTING MARKS make sure ALL and HIDDEN TEXT are UN-checked

However, my titles, footnotes, and class variable formatting was still off. The titles were resolved by adding formatting information to the title statements. I tore my hair out a little (or a lot) on the class variable issue but eventually resolved it by adding a CLASSLEV statement with font information. Hurrah! Despite my initial dismay, it is clear that SAS has given us even MORE control over every aspect of our ODS output in 9.1. You just have to know what to tell it!

It should be noted that using a user-defined style template in Version 9.1 might have resolved these issues. However, the purpose of this paper was to highlight PROCEDURE LEVEL styles. I will leave the wonders and mysteries of PROC TEMPLATE to other authors!

Example of Table 21 produced using SAS® 9.1 fixing MS Word® - Gobbledygook gone BUT...

Example of Table 21 produced using SAS® 9.1

Table 21-Unweighted
 Percentage of Schools Collecting Various Types of Data),
 By Selected School Characteristics:

	Has Access to Data	Has Access to Data	Types of Problems Presented by Student
TOTAL	649	100.0%	
Region			
Northeast	137	100.0%	
South	150	100.0%	
Midwest	228	100.0%	

**Table 21-Weighted
 Percentage of Schools Collecting Various Types of Data
 By Selected School Characteristics: 2002-2003**

	Has Access to Data		Types of MH Problems Presented by Students	Sc M
TOTAL	40,803	100.0%	66.8%	
Region				
Northeast	8,226	20.1%	65.6%	
South	10,516	25.7%	68.2%	
Midwest	13,865	33.9%	68.4%	
West	8,196	20.0%	63.7%	
Level				
Elementary	24,663	60.4%	64.0%	

Did you notice? There's another difference between the 8.2 and 9.1 tables. SAS ODS output to RTF automatically places titles and footnotes in the header and footer sections of MS Word® documents. You can prevent this from happening by specifying BODYTITLE at the end of your ODS RTF statement. In Version 8.2, you could not also specify NODATE NONUMBER to suppress the date and page number in your options along with BODYTITLE. In Version 9.1, this has been fixed. The titles are now in the body of the document, and not the header.

CONCLUSION

The use of PROC TABULATE with procedure level styles in conjunction with the RTF output destination allows the SAS programmer to produce complex tables with highly controlled formatting that can be read directly into Microsoft Word. (While reading into OpenOffice has also been tested, the text rotation technique does not work at the time of publication of this paper.) The ease of table production using this method greatly adds to the SAS programmer's toolbox. It also cannot be emphasized enough that being able to reproduce professional, accurate output without error-prone interventions such as copying and pasting is vital to the SAS programmer's repertoire.

REFERENCES

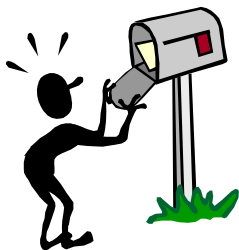
- SAS Online Documentation (PC SAS V8.2, AIX UNIX SAS V8.2, PC SAS V9.1)
 SAS Institute Inc., *The Complete Guide to the SAS® Output Delivery System*, Version 8, Cary, NC: SAS Institute Inc., 1999. 310 pp.
 Lauren E. Haworth, *PROC TABULATE by Example*, Cary, NC: SAS Institute, 1999. 392 pp.
 Cindy Tong, *ODS RTF: Practical Tips*, NESUG 2003
<http://support.sas.com/rnd/base/topics/odsrtf/rtf901.html>
http://support.sas.com/rnd/base/topics/templateFAQ/Template_rtf.html
 Microsoft Corporation; Rich Text Format (RTF) Specification, version 1.6
<http://msdn.microsoft.com/library/>



ACKNOWLEDGMENTS

SAS® Technical Support and R&D have been incredibly helpful, especially Wayne Hester and Bari Lawhorn. K.P. Srinath of Abt Associates Inc., statistical sampling and SUDAAN guru. Carolyn Robinson of Abt Associates Inc., whose work with RTF and PROC REPORT led to many insights. Cindy Tong, author of "ODS RTF: Practical Tips". Lauren Haworth, author of the definitive book "PROC TABULATE By Example". WUSS 2003 Code Clinic Staff. 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 registered trademarks or trademarks of their respective companies.

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
 Work Phone: 617-349-2385
 Fax: 617-349-2675
 Email: louise_hadden@abtassoc.com

KEYWORDS: SAS; ODS; RTF; PROC TABULATE; STYLE; TAGSET

Appendix A: Sample Program

```

options ps=48 ls=132 errorabend compress=yes;

libname dd '!';
filename odsout '!';

title1 'MHSCHOOL';
footnote "Program: C:\MHSCHOOL\DISTRICT\DTABLE5B.SAS -
last run &sysdate9.";
run;

proc format;
  picture pctpic low-high='009.9%';
  value loccat 1='Urban'
           2='Suburban'
           3='Rural';
  value region 1='Northeast'
              2='South'
              3='Midwest'
              4='West';
  ...
run;

data anal;
  set dd.distri02;
  ...
run;

options nocenter missing=' ' orientation=landscape;

%macro do_one(source=xxx);

data temp ;
  length line $400;
  infile &source length=lg lrecl=1000 end=eof;
  input @1 line $varying400. lg;

data _null_;
  set temp ; retain mod 0 ;
  file &source;
  /* mods are done for the trow header data only */
  if (index(line, "\trowd\trkeep") > 0) then do ;
    if (index(line, "\trowd\trkeep\trhdr") > 0) then mod = 1 ;
    else mod = 0 ;
  end;
  if (mod) then do;
  /* this works because they are the same width */
  line=tranwrd(line, "\cltxlrb", "\cltxbtlr") ;
  line=tranwrd(line, "\clvertalb", "\clvertalt") ;
  line=tranwrd(line, "\qc", "\ql") ;
  put line;
  /* The header cell needs to be tall enough to accept the
change. */
  /* You may need to adjust this depending on what your col-
umn headers are. */
  /* Measurment is in twips. 20 twips to the point, 72 points to
the inch.*/
  if ( index(line, "\trowd\trkeep\trhdr") > 0) then put "\trrh-1625"
;
  end;
  else put line ;
run;

%mend;

filename rtf 'wdtable5b.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",8pt,Normal)};
  var counter mjr_idea mjr_tit1 mjr_cbgr mjr_tit4 mjr_safe
  mjr_sped
      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",8pt,Bold)};
  weight basewgt3;
  class region loc_cat distsz3 /
  s={bordercolor=black borderwidth=2 font=("Times New Ro-
man",8pt,Bold)};
  tables all region loc_cat distsz3,
  counter*(sum=' '*f=comma9.)
  mjr_idea*(pctsum<counter>=' '*f=pctpic.)
  mjr_tit1*(pctsum<counter>=' '*f=pctpic.)
  mjr_cbgr*(pctsum<counter>=' '*f=pctpic.)
  mjr_tit4*(pctsum<counter>=' '*f=pctpic.)
  mjr_safe*(pctsum<counter>=' '*f=pctpic.)
  mjr_sped*(pctsum<counter>=' '*f=pctpic.)
  mjr_sgen*(pctsum<counter>=' '*f=pctpic.)
  mjr_ttax*(pctsum<counter>=' '*f=pctpic.)
  mjr_lott*(pctsum<counter>=' '*f=pctpic.)
  mjr_locl*(pctsum<counter>=' '*f=pctpic.)
  mjr_mcd*(pctsum<counter>=' '*f=pctpic.)
  mjr_3rdp*(pctsum<counter>=' '*f=pctpic.)
  mjr_schi*(pctsum<counter>=' '*f=pctpic.)
  mjr_self*(pctsum<counter>=' '*f=pctpic.)
  mjr_foun*(pctsum<counter>=' '*f=pctpic.)
  mjr_othr*(pctsum<counter>=' '*f=pctpic.)
  / box={s={bordercolor=black borderwidth=2}};
  keyword all / s={font=("Times New Roman",8pt,Bold)};
  keylabel all='TOTAL';
title1 'Table 5b-Weighted';
title2 'Percentage of School Districts For Which the Source is
Among Their Top
Five Sources';
title3 'By Selected District Characteristics: 2002-2003';
footnote1 'Source: Survey of the Characteristics and Funding of
School
Mental Health Services.';
footnote2 '2002-2003, Center for Mental Health Services, U.S.
Department
of Health and Human Services.';
footnote3 'District Questionnaire, Item 8';
run;

ods rtf close;

%do_one(source=rtf) ;

ods rtf close;

```

Appendix B – Sample Table

Table 5b-Weighted
 Percentage of School Districts For Which the Source is Among Their Top Five Sources,
 By Selected District Characteristics: 2002-2003

	# of Districts	Q8: IDEA Source of MH Svc Funding	Q8: Title 1 Source of MH Svc Funding	Q8: Comm MH Svcs Block Grant Source of MH Funding	Q8: Title 4 Source of MH Svc Funding	Q8: Safe Schools/Healthy Students Init Src of MH	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	Q8: SCHIP Source of MH Svc Funding	Q8: Self-Pay Source of MH Svc Funding	Q8: Priv Foundation Grants Source of MH Svc Funding	Q8: Other Source of MH Svc Funding 1
TOTAL	14,752	61.5%	22.3%	6.9%	52.8%	16.1%	51.5%	40.1%	5.8%	0.8%	47.1%	28.4%	4.9%	0.8%	7.5%	2.5%	10.0%
Region																	
Northeast	3,110	77.0%	17.0%	3.7%	58.2%	24.8%	54.6%	33.9%	0.8%	1.2%	71.6%	31.7%	3.7%	0.0%	7.7%	3.8%	6.9%
South	5,441	54.0%	26.1%	5.6%	53.2%	5.5%	52.7%	43.8%	2.7%	0.8%	50.0%	31.6%	4.8%	0.2%	8.1%	1.4%	9.4%
Midwest	3,330	61.3%	22.1%	7.2%	47.9%	21.3%	54.8%	42.0%	2.9%	0.3%	39.4%	26.3%	7.1%	3.2%	5.7%	1.4%	5.7%
West	2,872	59.4%	21.0%	12.1%	52.2%	21.0%	41.9%	37.6%	20.6%	1.1%	23.9%	21.2%	4.0%	0.0%	8.3%	4.3%	19.6%
Urbanicity																	
Urban	2,003	64.8%	35.1%	5.1%	47.5%	19.7%	57.8%	38.8%	9.0%	0.7%	38.5%	28.3%	4.5%	0.0%	6.4%	7.7%	12.2%
Suburban	4,773	64.8%	16.7%	6.0%	54.9%	14.4%	55.7%	38.1%	6.4%	1.7%	54.6%	32.4%	3.8%	1.0%	4.8%	2.9%	11.7%
Rural	7,976	58.7%	22.4%	7.8%	53.0%	16.3%	47.3%	41.7%	4.7%	0.3%	44.7%	26.0%	5.7%	0.9%	9.4%	0.9%	8.5%
District Size																	
1-5 Schools	10,651	56.9%	24.5%	5.1%	52.4%	15.6%	50.4%	38.9%	6.1%	0.2%	47.1%	25.4%	3.1%	0.6%	7.8%	2.3%	9.9%
6-15 Schools	3,100	73.0%	13.5%	10.0%	53.3%	15.4%	52.6%	43.0%	4.3%	2.4%	49.5%	36.1%	10.1%	1.2%	7.3%	2.6%	10.7%
16+ Schools	1,001	75.8%	25.7%	15.1%	56.5%	23.7%	59.2%	43.9%	8.2%	2.8%	39.3%	36.2%	8.0%	1.2%	4.7%	3.1%	9.6%

Source: Survey of the Characteristics and Funding of School Mental Health Services: 2002-2003, Center for Mental Health Services, U.S. Department of Health and Human Services, District Questionnaire, Item 8