

Paper 059-2009

PROC REPORT: Titling It Right, with Footnotes to Boot

William C. Murphy

Howard M. Proskin & Associates, Inc., Rochester, NY

ABSTRACT

Using PROC REPORT, you can generate a variety of summary tables. When these tables are created in Rich Text Format via ODS, the output is nearly publication-ready. Nonetheless, attention must be paid to the alignment, position, and number of lines of the titles and footnotes. Using the TITLE and FOOTNOTE statements, such text is positioned with the top and bottom margin of the output page, respectively. Left, right, or a combination alignment is readily achieved with the justification option. To position and align titles and footnotes with the table boundaries, however, the text should be incorporated into the PROC REPORT code. This can be done by judiciously inserting text in the COLUMN statement and by use of the LINE statement in a compute block. Furthermore, such text is not limited to the 10 lines of the TITLE and FOOTNOTE statements. Both methods of generating titles and footnotes require special consideration of spaces and blank lines.

OVERVIEW

Your program works perfectly. Using PROC REPORT, it puts the desired results into the appropriate cells in a table in the output. Now you route the output of your program via ODS into an RTF document. The margins and fonts are fine. All that you need to do is add some titles and footnotes. This should be very straightforward: just use the SAS® TITLE and FOOTNOTE statements.

You examine the template provided by the client. Some of the titles are aligned on the left margin, some on the right, and some on both margins! Every title line seems to be different. The footnotes are no better. The client wants some of them to line up with the lower part of the table and others with the page margin. What's worse you notice that some of the tables have more than ten title and/or footnote lines. Doesn't SAS have a ten-line limit?

All of these problems can be readily solved. In the following discussion, we will demonstrate solutions using the appropriate options on the TITLE and FOOTNOTE statements. We will also show that it is preferable to relegate some of the titles and footnotes to the PROC REPORT code.

SET-UP

You are requested to summarize the age, weight, and height of your study population by sex. So you write the following code:

```
proc report data=sashelp.class nowindows
  style=[rule=group frame=void];

  column sex age height weight;

  define sex/group;
  define age/mean;
  define height/mean;
  define weight/mena;
run;
```

where we have used the RULE and FRAME options to remove the grid in and around the table, except below the column headers. The code produces exactly the table that you want. So now you simply add two statements before this code to produce your title and footnote:

```
title1 'Demographics';
footnote1 'The subjects in the table are from a sample population';
```

Now your client gives you some changes. He wants his company name on the left side of the first title line and the project name on the right side of the same line. On the second title line, he just wants the project number on the right side. Then he wants the title 'Demographics' centered on the next line followed by a space and then the line 'of the Selected Population'. He would also like the footnote immediately under the table and aligned with the left side of the

table. In addition, he wants to append an extremely long explanatory footnote to the end of the table, followed by a footnote at the extreme bottom of the page with the program name, date and time of execution.

ALIGNMENT

If you are generating the output to the SAS Output window, the SAS system option CENTER/NOCENTER allows you to align the titles and footnotes to the center or left, but you must align all title and footnotes in the same way. Furthermore, all the text and tables generated will also be aligned in this fashion. The tedious way of fixing this problem is by choosing CENTER and then by padding the text in titles and footnotes with spaces to achieve the proper justification. Now SAS has a marvelous way of ignoring spaces at the beginning of the text in TITLE and FOOTNOTE statements. However, this annoyance can be overcome by using a macro quoting function containing the spaces. For instance, our second title line could be written

```
title2 "%str(                )Project #31415";
```

This code will left align the second title if you insert enough space in the %STR function. Alternately you could try to use a null character instead of a space. A null character is obtained by holding down the ALT key and typing 255 on the keypad (not the digits above the qwerty row). However, you should be warned that in some fonts this key combination will produce a symbol rather than a space.

For nearly all of the reports that we generate, we use ODS to write to a Rich Text Formatted (RTF) file. So we start our reporting program with

```
ods rtf file='z:\projects\demog.rtf' style=minimal;
```

Now in RTF output, the CENTER/NOCENTER option works exactly like in does for the output window and so does the null character trick. The macro quoting of lead spaces, however, does not work. Fortunately, ODS allows us a more direct approach to justify titles. We can use the justification option to right align the second title:

```
title2 j=r 'Project #31415';
```

This will readily align the second title to the right without using any funky spaces.

Now let's go back to the first title line. Remember the client wants his company name on the right and the project name on the left. We could write this title with lots of spaces between its two parts or we could write

```
title1 j=l 'The Prolix Company Unlimited'
        j=r 'The Verbose Project';
```

and our first title would come out just perfect. SAS can us multiple justification options on both its TITLE and FOOTNOTE statements. I could have even entered text in the center using j=c if needed. This solved our problem for the first two titles that the client wanted, but what about the other titles?

POSITION

The client would like the next title lines written adjoining the table and not immediately under the first two titles. If we continue to use TITLE statements, the output will be associated with the previous titles and not with the table.

The PROC REPORT procedure allows us to write text to span the headers that it outputs for the columns. If we write text to span all of the columns, we will in effect be placing a title across the entire table. Rewriting the column statement from above, we get

```
column ("Demographics" sex age height weight);
```

where the parentheses delimit the range of the spanning text, in this case, all columns. But don't forget the client wants a second title on the table and some blanks lines between them. So we write

```
column ("Demographics" " " "of the Selected Population" " " sex age height
        weight);
```

Note that if you list text strings one after the other, PROC REPORT will stack them on top of each other. So this code should produce our desired output. And it does in the SAS Output windows. However, our RTF output is missing the blank lines!

BLANK LINES

I don't know why the SAS RTF interfaces gets rid of blank lines, but I do know that if the client wants blank line inserted, they must be inserted. If you try our previous trick using a macro quoting function to inset blanks, it will not work. The problem has nothing to do with blanks per se, but rather with the interface. The Alt-255 trick will work but the same *caveat* as above applies. However, we can insert an RTF blank in the text. First we must choose an ODS escape character:

```
ods escapechar='^';
```

where we have chosen the caret because it's unlikely to appear in our output. With the escape character defined, we can readily invoke a RTF space by writing '^R' ". So we can rewrite our column statement as

```
column ("Demographics" '^R' "" "of the Selected Population" '^R' "" sex age
        height weight);
```

This will produce the desired titles in our RTF title, although the SAS Output window now has extraneous characters where blank lines should be.

FOOTNOTES

So we have solved our title problems, but what about the footnotes. Most of what was said about the titles applies to the footnotes. To get the footnotes that we want at the extreme bottom of the page we use the FOOTNOTE statement:

```
footnotel j=1 'Demog.sas run on 31 September 2008 at 12:12 am';
```

which will give us a nice left aligned text..

Just as with the titles, we include footnotes that we want to associate with the table in the PROC REPORT code. There is no such thing as text spanning the bottom of the columns, but the same effect can be accomplished by using a COMPUTE block at the bottom of our PROC REPORT code:

```
compute after/style=[just=r];
  line ' ';
  line 'The subjects in the table are from a sample population';
endcomp;
```

This will produce our footnote aligned with the right side of the table, not the margin. If the text was extremely long it would wrap at the left side of the table. This is exactly what the client wants. The LINE statement code for the blank line actually works in both the SAS Output window and the RTF document!

Remember that the client wanted to add an extremely long footnote. We can put this in another LINE statement before the ENDCOMP:

```
line '...the encyclopedia ...';
```

As long as there is room on the page, SAS will insert this footnote, keeping it within the left and right margin of the table. There is no ten-line limit as there is for the FOOTNOTE statement.

WARNING

If your footnote gets too long SAS might write this warning to the LOG,

```
WARNING: 32-169 The quoted string currently being processed has become
              more than 262 characters long. You may have unbalanced
              quotation marks.
```

From my experience, this warning has no effect on your output and can be ignored. But it is still unpleasant to have in your log. However, you can suppress this warning by using a system option before running the PROC REPORT:

```
option noquotelenmax;
```

I would strongly suggest using this code only after the bugs are worked out. It could actually be warning you of a missing quote and not just of your ridiculously long footnote.

Also if your LINE statement text is more than 960 characters, SAS will be unable to process it. However, you can divide the text into separate strings, ending the strings at places where you would expect a carriage return. These separate strings could then be written with multiple LINE statements. After all this, if your wordy footnote is too long to fit on the output page, you may need to adjust the vertical line size to make everything fit. To do this we could write:

```
proc report data=sashelp.class nowindows
  style=[rule=group frame=void]
  style(column)=[cellheight=4 mm];
```

However, if you have to make your line size less than 4 mm for the footnote to fit on the page, your font size should be less than 10-point or some of the characters may have their tops or bottoms clipped. If all this fails, it may be time to consider editing the footnotes or redesigning the template.

CONCLUSION

Using PROC REPORT, along with the SAS TITLE and FOOTNOTE statements, you can produce output in an RTF file that has titles and footnotes exactly as the client desires. By using spanning headers and LINE statements, titles and footnotes can be directly associated with the PROC REPORT table output. Using the justification options and the occasionally RTF space, output made with TITLE and FOOTNOTE statements can be aligned and spaced just right.

APPENDIX

The working program developed here, in final form, is

```
title1 j=l 'The Prolix Company Unlimited'
      j=r 'The Verbose Project';

title2 j=r 'Project #31415';

footnote1 j=l 'Demog.sas run on 31 September 2008 at 12:12 am';

option noquotelenmax;

ods escapechar='^' ;

ods rtf file=' z:\projects\demog.rtf ' style=minimal;

proc report data=sashelp.class nowindows
  style=[rule=group frame=void]
  style(column)=[cellheight=4 mm];

  column ("Demographics" '^R' "" "of the Selected Population" '^R' "" sex age
    height weight);
  define sex/group
    style=[CellWidth=20%];
  define age/mean
    style=[CellWidth=20%];
  define height/mean
    style=[CellWidth=20%];
  define weight/mean
    style=[CellWidth=20%];

  compute after/style=[just=1];
    line ' ';
    line 'The subjects in the table are from a sample population';
    line ' ';
    line 'Many, many, many, many, many, many, many, many, many, words
    . . . the encyclopedia . . . many, many, many, many, many, many, more
    words';
  endcomp;

run;
```

```
ods rtf close;
```

The data set CLASS is a sample data set supplied with your SAS software and is found in the SASHELP library.

CONTACT INFORMATION

Your comments and questions are values and encouraged. Contact the author at

William C. Murphy

Howard M. Proskin & Associates, Inc.

300 Red Creek Dr., Suite 220

Rochester, NY 14623

Phone 585-359-2420

FAX 585-359-0465

Email wmurphy@hmproskin.com or wcmurphy@usa.net

Web www.hmproskin.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 registered trademarks or trademarks of their respective companies.