Paper 79

SUGI 24

User-Specified Text Flow Inside The Report Procedure

Donald W. Peterson, PSCI – Progressive Software Computing Inc., Wilmington, DE
John R. Gerlach, Maxim Group, Plymouth Meeting, PA

Abstract

The FLOW option of the Report procedure in SAS® allows the user to include lengthy text in a report by wrapping the text within a specified column width. This powerful feature splits the text using the so-called split character, as well as the blank character. The text is always left justified, not allowing any kind of indentation, unless you insert blank characters accordingly, which is very tedious. This paper explains the **%flowvar** macro that modifies the character variable allowing the FLOW option to produce indented text, beginning after the first line for each observation in a report.

Introduction

The Report procedure, made available in Version 6, superseded the DATA _null_ step and introduced the SAS user to new ways to produce sophisticated reports. Such features as using ACROSS variables that become column headers and generating descriptive statistics that become part of the report exemplify the power of the Report procedure, especially in contrast to previous techniques. Although the DATA _null_ step allows complete control over the layout of the report, the Report procedure can handle virtually any well-designed report. (Notice the word well-designed).

One of many powerful features found in the Report procedure includes the FLOW option. The FLOW option wraps the text of a character variable within its column, using the split character, e.g., the slash (/). If the text contains no such character, the procedure attempts to split the text based on natural word boundary, that is, the blank character. The result is left-justified text that fits within a specified column.

The Flow Option

Beginning with a DATA step that creates a contrived data set, the following code illustrates the FLOW option found in the Report procedure, including output.

```
data rep;
   input @4 id sex $ age @11 comment $60.;
cards;
1 M 40 xxxxxxxxxx xxxxx xx xx xxx x xxxxxxxx
2 F 43 xxxxxxx xx xxxx xxxxx xxxxxxxx xxxxxxxx
3 F 45 xxx xxxx xx xxxx xxxxx xxxxxxxx xxxxxxxx
;

proc report data=rep;
   column id sex age comment;
   define id      / order;
   define sex     / display width=3;
   define age     / display width=3;
   define comment / display flow width=10;
   break after id / skip;
   title 'Using FLOW Option Without Adjustment';
run;
```

The Report procedure produces the output shown below.

```
Using FLOW Option Without Adjustment

ID  SEX  AGE  COMMENT
 1  M     40  xxxxxxxxxx
              xxxxx xx
              xx xxx x
              xxxxxxxx

 2  F     43  xxxxxxx
              xx xxxx
              xxxxx
              xxxxxxxx
              xxxxxxxx

 3  F     45  xxx xxxx
              xx xxxx
              xxxxx
              xxxxxxxx
              xxxxxxxx
```

Notice how the procedure affords as many lines as necessary to fit the text variable in the column, specified by the WIDTH option. Also, notice that the text is left-justified. The %flowvar macro takes this feature a step further by indenting the text beginning after the first line. Actually, the macro modifies the contents of the character variable affecting how the FLOW option processes it.

The %FLOWVAR Macro

The %flowvar macro greatly enhances the FLOW option by adjusting the text variable prior to its use. The macro consists of four positional and two keyword parameters. The positional parameters should be specified always; whereas the others have default values. Besides knowing the name of the input and output data sets, the macro must know the name of the text variable that needs adjusting and how to adjust it, that is, the target column width, the indentation, and the split character. The options are as follows:

indsn	Name of input SAS data set
outdsn	Name of output SAS data set
flowvar	Name of text variable to be indented
width	Width of column used in Proc Report
indent	# columns indented after first line
split	Character used to split text

The %flowvar macro, shown below, begins by assigning the parameter *outdsn*, denoting the output data set, the value of *indsn*, the name of the input data set, in the event that the user does not specify a name for the output data set. Although this option does not prevent an unwanted overwrite of the input data set, it does allow repeated use of the macro affecting more than one character variable in the same data set, prior to reporting.

The remaining portion of the macro consists of a single DATA step that modifies the text variable, indicated by the *flowvar* option, so that the FLOW option can format the text in a more preferred style. Basically, for every observation, the macro processes the specified variable and inserts a split character (e.g., the slash) accordingly, at appropriate places. To accomplish this task requires three DO WHILE loops and a rigorous use of the SUBSTR function. The DO loops adjust a pointer variable in search of an appropriate place for the split character. The SUBSTR function inserts the split character using the function on both sides of the assignment statement. Finally, the TRANWRD function does the final adjustment by translating the split character (treated as a word) into a word appended with blanks, thereby inserting the appropriate indentation after the first line of text.

```
%macro flowvar(indsn,outdsn,flowvar,width,
   indent=2,split=/);
   %if &outdsn. eq
      %then %let outdsn = &indsn.;
   data &outdsn.;
      length indent $&indent. &var. $200.;
      retain indent z_ptr;
      set &indsn.;
      ptr = &width. + 1; z_ptr = &width.;
      do while(substr(&flowvar., ptr,1) ne ‘’);
         ptr = ptr - 1;
         if ptr eq 0 then do;
            %cuthalf;
         end;
      end;
      substr(&flowvar.,ptr) = “ &split.”  ||
         substr(&flowvar.,ptr+1);
      do while(ptr lt length(&flowvar.));
         ptr = ptr + (&width.-&indent.) + 1;
         z_ptr = ptr-1;
         do while(substr
            (&flowvar.,ptr,1) ne ‘’);
            ptr = ptr - 1;
            if z_ptr-ptr eq &width.-&indent.
            then do;  %cuthalf;  end;
         end;
         substr(&flowvar.,ptr) = “ &split.”  ||
            substr(&flowvar.,ptr+1);
      end;
      &flowvar. = tranwrd(&flowvar., “ &split.” ,
         “ &split.” ||indent);
      drop ptr z_ptr indent;
   run;
%mend flowvar;
```

```
%macro cuthalf;
 z_var = substr(&var.,1,z_ptr-1) || “ - “   ||
            substr(&var.,z_ptr);
    &var. = z_var;
    drop z_var;
    ptr = z_ptr + 1;
 %mend cuthalf;
```

Examples

The following two examples illustrate proper use of the %flowvar macro.

```
%flowvar(rep,newrep,comment,10,indent=2,split=!)
```

The first example adjusts the text variable COMMENT so that the FLOW option wraps its contents within a column width of ten spaces and indents all but the first line with two spaces. This invocation creates a revised data set called NEWREP that adjusts the variable COMMENT. Also, the exclamation mark supplants the default split (i.e., slash) character.

After adjusting the variable COMMENT, the Report procedure processes the character variable, using several required options. In this example, the SPLIT option of the PROC statement indicates the same character used by the %flowvar macro. Also, the FLOW and WIDTH options of the DEFINE statement process the adjusted character variable as intended.

```
proc report data=newrep split=’!’;
   column id sex age comment;
   define id      / order;
   define sex     / display width=3;
   define age     / display width=3;
   define comment / display flow width=10;
   break after id / skip;
   title ‘Enhanced Text: Width=10, Indent=2’;
run;
```

The Report procedure produces the output shown below. Notice how the Report procedure processes the variable COMMENT after using the %flowvar macro.

```
Enhanced Text: Width=10, Indent=2


ID  SEX  AGE  COMMENT
 1  F     40  xxxxxxxxxx
                xxxxx xx
                xx xxx x
                xxxxxxxx

 2  M     43  xxxxxxx xx
                xxxx
                xxxxx
                xxxxxxxx
                xxxxxxxx

 3  M     45  xxx xxxx
                xx xxxx
                xxxxx
                xxxxxxxx
                xxxxxxxx
```

The next example adjusts the same character variable, expanding the column width to twice the size and increasing the indentation slightly.

```
%flowvar(rep,rep,text,20,indent=3,split=!)
```

```
proc report data=rep split=!;
   column id sex age comment;
   define id      / order;
   define sex     / display width=3;
   define age     / display width=3;
   define text    / display flow width=20;
   break after id / skip;
   title ‘Enhanced Text: Width=20, Indent=3’;
run;
```

The Report procedure produces the output shown below. Notice how the procedure processes the adjusted variable, this time indenting three spaces after the first line for each observation and using a wider column width.

```
Enhanced Text:  Width=20, Indent=3

ID  SEX  AGE  COMMENT
 1  F     40  xxxxxxxxxx xxxxx xx
                 xx xxx x xxxxxxxx

 2  M     43  xxxxxxx xx xxxx
                 xxxxx xxxxxxxx
                 xxxxxxxx

 3  M     45  xxx xxxx xx xxxx
                 xxxxx xxxxxxxx
                 xxxxxxxx
```

The Report procedure produces the output shown below. Notice how the procedure processes the adjusted variable, this time indenting one space after the first line for each observation and using a short column width then the designated width of the column specified in parameter.

```
Enhanced Text:  Width=9, Indent=1

ID  SEX  AGE  COMMENT
 1  F     40  xxxxxxxx-
               xx xxxxx
               xx xx
               xxx x
               xxxxxxxx

 2  M     43  xxxxxxx
               xx xxxx
               xxxxx
               xxxxxxxx
               xxxxxxxx

 3  M     45  xxx xxxx
               xx xxxx
               xxxxx
               xxxxxxxx
               xxxxxxx-
               xx
```

Assumptions

The %flowvar macro makes the following assumptions:

- A blank character denotes a word boundary.

- The input data set does not contain variables named PTR, Z_PTR or INDENT.

Conclusion

The FLOW option found in the Report procedure formats character (text) variables so that it fits in a pre-defined column width, in a left-justified manner. The %flowvar macro exploits this option by adjusting the character variable so that its contents is indented after the first line of each observation, thereby producing a more desirable report.

Author Information

Donald W. Peterson
PSCI - Progressive Software Computing Inc.
3505 Silverside Road, Suite 201-B
Wilmington, DE 19810

302.479.9700   Fax: 302.479.7573

E-mail: DONALD_PETERSON@MERCK.COM
DPETERSON@PSCI.COM

John R. Gerlach
183 Freeland Drive
Collegeville, PA 19426
610.454.9967

E-mail: GERLAJ01@IMSINT.COM