A Productive Word Wrap Macro

Monte Jarvis, Genzyme Inc
Frank Liu, Genzyme Inc

INTRODUCTION:

If a data set contains a large number of variables, and one needs to print full observations with some wrapped variables on each page, PROC SQL and PROC REPORT cannot handle the job although they both have flow options. This paper presents a productive SAS macro using a *Split-Merge* approach to wrap as many long text variables as required. Generally the wrap macro can be implement before using PROC PRINT. The output is neat and readable.

THE SPLIT-MERGE APPROACH

When a long text variable needs to be wrapped, normally, this variable can be split into lines based on a given output length, and then the split text can be written out to a SAS data set. However, the SAS data set will include the duplicated value of non-wrapped variables. To avoid this duplication, you can *split* an input data set into two output data sets. The first data set contains the first split line of the long text variables. The second data set contains the remaining split lines of the long text variables and other variables. Because both data sets also include a run-time variable ctr about the order of observation, therefore, *merging* these two data sets by the variable ctr will avoid the duplicated value of non-wrapped variable.

THE WRAP MACRO: WORDWRAP

WORDWRAP is invoked with four keyword parameters. These keywords, used throughout the macro in the form of macro variables, enable the user to achieve a great deal of flexibility from using this productive macro: one can choose any text variables for wrapping, and one can wrap as many long text variables as required.

Here is the text of the macro with a detailed explanation:

```
*************************************;
*                                   *;
*  Parameters Specifications        *;
*  -------------------------        *;
*                                   *;
*  data Input data set name.        *;
*  id   ID variable name. ID        *;
*       variable can be one or more, *;
*       such as, id=id1, id2, id3... *;
*  var  Long text variable name.    *;
*       This var can be one or more, *;
*       such as, var=var1 var2 var3..*;
*  len  Width of each line of       *;
*       wrapped variables. Len      *;
*       contains different lengths  *;
*       corresponding to each long  *;
*       text variable, such as,     *;
*       len=len1 len2 len3...       *;
*                                   *;
*************************************;

%MACRO WORDWRAP
   (data=, id=, var=, len=);

*************************************;
*                                   *;
* Description of Macro Variables    *;
* ------------------------------    *;
*                                   *;
* _n   Number of the ID variable.   *;
* _m   Number of the long text      *;
*      variables.                   *;
*_com  Hold _COM1, _COM2, _COM3...  *;
*_blk  Hold _BLK1, _BLK2, _BLK3...  *;
*                                   *;
*                                   *;
*************************************;

 %let _n=1;
 %let _m=1;
 %let _com=;
 %let _blk=;

 %do %until (%scan(&id,&_n)=);
    %let  id&_n=%scan(&id,&_n);
    %let  _n=%eval(&_n+1);
 %end;
 %let _n=%eval(&_n-1);
       /* Get the number of the id
          variables                  */
```

```
 %do %until (%scan(&var,&_m)=);
     %let  var&_m=%scan(&var, &_m);
        /* Get var1, var2, var3 ... */

     %let  len&_m=%scan(&length,&_m);
       /* Get len1, len2, len3...  */

     %let  _m=%eval(&_m+1);
 %end;

%let _m=%eval(&_m-1);
       /* Get the number of the long
          text variables            */

 %do _i=&_m %to 1 %by -1;
   %let _com=_COM&_i &_com;
   %let _blk=_BLK&_i &_blk;
 %end;

*************************************;
*                                   *;
* SUB1 contains the first split line *;
*      of long text variables.      *;
*                                   *;
* SUB2 contains the remaining split  *;
*      lines of long text variables. *;
*                                   *;
* Description of Variables          *;
* ------------------------          *;
*                                   *;
* _COM1, _COM2, ...                 *;
*      Store the remaining text of  *;
*      the long text variables after *;
*      split line.                  *;
* _BLK1, _BLK2                      *;
*      Word break point at end line. *;
* CTR  Counter for observation to be *;
*      Written to a SAS data set.   *;
* _L   split line number of the long *;
*      text varlabes.               *;
*************************************;

DATA SUB1(DROP=&_com &_blk _J _L)
     SUB2(KEEP=&id &var CTR);
 LENGTH &_com $200;
 RETAIN CTR 0;
   SET &data;
   _L=1;
 RUN;

*************************************;
*                                   *;
* PREWRAP:                          *;
* -------                           *;
*                                   *;
* 1.  Determines whether the long   *;
*     text variable need to be      *;
*     wrapped or not.               *;
* 2.  _L =1 indicates that the all  *;
*     long text variables of current *;
*     observation do not need to be *;
*     wrapped.                      *;
* 3.  _L > 1 indicates some or all  *;
*     long text variables have been *;
*     successfully wrapped.Last line *;
*     of long text variables remains.*;
*                                   *;
*************************************;

PREWRAP:

   %do _i=1 %to &_m;
     _COM&_i=&&var&_i;
   %end;
   %do _i=1 %to &_m;
     IF length(&&var&_i)>&&len&_i THEN
        GOTO WRAP;
     ELSE
       IF &_i=&_m AND _L=1 THEN
     DO;
        CTR+1;
        OUTPUT  SUB1;
     END;
     ELSE
      IF &_i=&_m AND _L>1 THEN
     DO;
        CTR+1;
        OUTPUT  SUB2;
     END;
   %end;
 RETURN;

*************************************;
*                                   *;
* WRAP:                             *;
* ----                              *;
* A wrap cycle starts with scanning *;
* the long text variable letter by  *;
* letter from right to left until   *;
* word break point is found at the  *;
* end of line.                      *;
*                                   *;
*************************************;

WRAP:

   %do _i=1 %to &_m;
      _BLK&_i=0;
      IF LENGTH(_COM&_i)>&&len&_i THEN
      DO;
        DO _J=&&len&_i TO 1 BY -1;
          IF SUBSTR(_COM&_i,_J,1)=' '
            /* Look backwords for a
               blank until a blank is
               found                */
          THEN
           DO;
              _BLK&_i=_J;
              GOTO LOOP&_i;
           END;
        END;
      END;
```

```
LOOP&_i:

  %end;

  %do _i=1 %to &_m;
  IF _BLK&_i ne 0 THEN
   &&var&_i=SUBSTR(_COM&_i,1,_BLK&_i);
  %end;
  IF _L=1 then
     DO;
        CTR+1;
        OUTPUT  SUB1;
     END;
   ELSE
     DO;
        CTR+1;
        OUTPUT  SUB2;
       /* if _blk ne 0, then you are
          in the middle of the word  */

     END;
   _L+1;
  %do _i=1 %to &_m;
   IF _BLK&_i NE 0 THEN
    &&var&_i=SUBSTR(_COM&_i,_BLK&_i+1);
   ELSE
    &&var&_i=' ';
  %end;
 GOTO PREWRAP;

DATA &data;
  SET SUB1 SUB2;
  BY CTR &id;
RUN;

%MEND WORDWRAP;
```

EXAMPLE:

The following program shows how to create a word-wrapped SAS data set by using WORDWRAP. The word-wrap data listing is shown in Appendix.

```
LIBNAME SSD '[PROJECT.STUDY.SSD]';

DATA A;
  SET SSD.ADVERSE;
    BY PATID;
  IF FIRST.PATID THEN AENUM=1;
  ELSE AENUM+1;
RUN;
```

```
%WORDWRAP
  (data=a, id=patid, var=ae comments, len=20 25);
          /* Length of var AE is 20 , and
              length of var comments is 25 */

PROC PRINT DATA=A;
  ID PATID;
  BY PATID;
 /*  When the ID and BY variables are the same,
      the ID variable is listed only once for each
    by group                                    */

RUN;
```

REFERENCES:

1. SAS Institute Inc., *SAS Guide to Macro Processing, Version 6, Second Edition,* Cary, NC: SAS Institute Inc., 1990.

2. SAS Institute Inc., *SAS Procedures Guide. Version 6, Second Edition,* Cary, NC: SAS Institute Inc., 1990.

CONTACT INFORMATION:

Monte Jarvis
Biostatistics & Medical Data Processing
Genzyme,Inc.
One Kendall Square
Cambridge, MA 02139

Tel: (617)252-7693
E-mail: mejarvis@world.std.com

Frank Liu
Biostatistics & Medical Data Processing
Genzyme,Inc.
One Kendall Square
Cambridge, MA 02139

Tel: (617)374-7262
E-mail: fliu@world.std.com

Appendix: An Example of Word-Wrapped Data Listing

GENZYME CORPORATION — STUDY NUMBER — AE
CAMBRIDGE, MASS. — PRODUCT DESCRIPTION — 14:56/18DEC95

ADVERSE EVENTS

PATID	AE ID	PRES AT BASELINE	NO AES	AE DESCRIPTION	HISTORY OF EVENT	SEVERITY	START DATE	STOP DATE	START TIME	TYPE OF EPISODE	DUR (MINS)	DUR (HRS)	DUR (DAYS)
901	1	Y		WITHDRAWAL SXS.	N	MOD	14MAY94	25MAY94	0:00:00	INTERMITTENT			11
	2	Y		FEVER	Y	MILD	15MAY94	20MAY94	17:00:00	INTERMITTENT			6
	3	Y		CELLULITIS	N	MILD	16MAY94	24MAY94	0:00:00	SINGLE			9
902	1	N		NAUSEA	N	MILD	16MAY94	17MAY94	18:00:00	INTERMITTENT			1
903	1	Y		FEVER	Y	MOD	24MAY94	24MAY94	16:00:00	INTERMITTENT	0	0	1
	2	Y		LEFT ARM CELLULITIS	N	MILD	25MAY94	31MAY94	6:45:00	SINGLE	0	0	6
	3	N		SHAKING CHILLS	N	MOD	25MAY94	25MAY94	7:30:00	SINGLE	35	0	0
	4	N		ARRHYTMIA	N	MILD	27MAY94	28MAY94	0:00:00	SINGLE			2
	5	Y		FEVER	Y	MOD	27MAY94	28MAY94	15:45:00	SINGLE	0	0	2
	6	N		CONSTIPATION	N	MILD	28MAY94	28MAY94	0:00:00	SINGLE	0	0	1
904	1	N		BACK ITCHING RASH NOTED	N	MILD	11JUN94	11JUN94	23:00:00	SINGLE			

PATID	ACTION TAKEN	OTHER ACTION	RELATIONSHIP TO PRODUCT	OUTCOME	AE COMMENT	-COSTART- COSTARTS TERM	-COSTART- SUBCATE TERM	-COSTART- BODY TERM
901	NONE	Y	NO	Rec	PATIENT RECEIVED METHODONE	WITHDRAW SYND	CNS/B	NER
	NONE	Y	NO	Rec	TREATED WITH TYLENOL	FEVER	GEN	BODY
	NONE	Y	NO	Rec	PATIENT TREATED WITH ANCEF	CELLULITIS	GEN	BODY
902	NONE	Y	NO	Rec	COMPAZINE AND MYLANTA	NAUSEA	GEN	DIG
903	NONE	Y	NO	Rec	FEVER TOOK TYLENOL.	FEVER	GEN	BODY
	NONE	Y	NO	Rec	ANTIBIOTIC	CELLULITIS	GEN	BODY
	NONE	Y	POSS	Rec	PATIENT MONITORED, BLANKET PROVIDED RESOLVED NO SEQUELAE.	CHILLS	GEN	BODY
	NONE	Y	NO	Rec	EKG AND MEDICAL CONSULT	ARRHYTHMIA	CARD/ARR	CV
	NONE	Y	NO	Rec	FEVER TOOK TYLENOL	FEVER	GEN	BODY
	NONE	Y	NO	Rec	LAXATIVE	CONSTIP	EC	DIG
904	NONE	Y	NO	Rec	ITCHING RELATED TO ELASTIC NETTING USED TO HOLD DRESSINGS IN PLACE.	RASH	DERM/ERY	SKIN

NA = NOT AVAILABLE ND = NOT DONE -99 = MISSING NUMERIC VALUE
N/A = NOT APPLICABLE UNK = UNKNOWN 01JAN01 = MISSING/PARTIAL DATE