

## USING THE SAS<sup>®</sup> FSLETTER PROCEDURE IN MACRO TO PRODUCE QUALITY LETTER OUTPUT FOR A SALES AND MARKETING DEPARTMENT

Phyllis D. Jordan-Johnson, *The Bureau of National Affairs, Inc.*  
Margaret L. Molnar, *The Bureau of National Affairs, Inc.*

### INTRODUCTION

The Bureau of National Affairs, Inc., located in Washington, D.C., is a publishing concern which provides information services to professionals in business, tax, legal, labor, environment and safety fields. The Sales and Marketing department has a data processing group, Marketing Information Systems, which uses the SAS<sup>®</sup> programming language in an MVS environment. SAS is used for the reporting and management of sales and marketing data essential in decision making. Several months ago, our unit began experimenting with the SAS FSLETTER product, as a means of merging subscriber files with advertising and promotion letters. With the aid of our Systems Support Department, we have been able to create formatted letters which satisfy a very discriminating copy department. The system has replaced an antiquated manual operation which was used to send personalized letters to prospective BNA customers who were receiving a BNA service on a free trial basis.

The letters generated are used to encourage those who take one of the BNA services on a forty-five day trial basis to subscribe to that service for a year or more. The system we developed, which is incorporated in a macro, has been in production since June of 1988. It has proven to be extremely useful, so much so that it is being considered as a possible replacement for some of the often expensive small personalized promotions which have always been generated out-of-house.

Our objective in this paper is to describe how we incorporated the FSLETTER procedure, several SAS functions, and SAS Macro System in the development a date-driven personalized letter system which allows the department to not only use the SAS language to manage its marketing database for analysis, but to generate real cost savings by producing quality mailing pieces in-house.

### DATABASE DESCRIPTION

The database used for the letter generation is housed in a SAS dataset on an IBM mainframe and is updated weekly. It comes from our online CICS customer database, managed by the Information Systems Development Department. The data is read into an upload file on tape from which our macro pulls the appropriate key, checks for duplicates and updates our letter file. The data is stored in upper case and is converted to mixed case in an algorithm described in the subsequent section.

### THE MACRO AND THE ALGORITHM

#### Macro

The system incorporates the use of SAS macros to retrieve the data weekly, reformat the name and address variables into mixed case (a refined letter mode), and print letters for customers whose forty-five day trial period has come to a close. The first macro, "Spirlet", retrieves the account information of the active records coded for the 45 day free trial we are tracking, and stores it in a temporary dataset. The second macro "Hletin" merges the name and address data with the temporary dataset. Following this, the macro "Advlet3" extracts the records that have had the free trial for more than 35 days, updates the permanent dataset and calls the macro "Mcase". "Mcase" converts the data to upper and lower case and prints the appropriate letter. See Appendix A for the macro code.

#### Algorithm

The Algorithm in "Mcase" uses a combination of the *Translate*, *Scan*, *Substr*, *Trim*, *Left*, and *Index* functions to convert the data to upper and lower case. The *Scan* and *Index* functions are used to separate each value (company name, address, etc.) into words. The *Substr* function is used to extract the first letter of the word so that the *Translate* function converts only the remaining portion of the word into lower case. The *Trim* and *Left* Functions are used to remove extra spaces when concatenating the words. (See Appendix B)

### FORMATTING QUALITY LETTERS USING FSLETTER

#### Carriage Control Commands and Fonts

Carriage control characters are used to designate the spacing throughout the letter. They are keyed in the first column of the document. The standard printer control characters used are:

+	Overprint(No skip)
0	Double space
-	Triple space
1	Top of page

Use of these characters enables us to use the fonts necessary for the various letter formats.(See Appendix C)

The number of fonts available to us at the present time, is ten. The font is indicated in the second column of the letter. (See Appendix C) Using this format, we are able to bold, use italics and underline.

In addition, the attributes of the variables used in the letter are given in Appendix D. The print file parameters, text body and margin information are found in Appendix E. A sample of the merged letter is found in Appendix F.

## HARDWARE AND OPERATING SYSTEM CONSIDERATIONS

### Hardware

Although letters can be produced on most IBM terminal with 24 lines and 80 character width per screen, we had to use the IBM 3278, Model 5 which allows 27 lines and 132 character width per screen. This was due to the fact that the copy was formatted on a Wang word processor and the integrity of the copy had to be maintained. The printers available to us are Xerox 8700 and Xerox 4050, mainframe printers.

### Operating System

The operating system used to create the letter file is ETSO emulated under Roscoe. The batch operation which merges and produces the letters, is MVS/Roscoe.

### Job Control Language

To queue the operator to change the paper to the letterhead required, the following JCL statement was used:

```
//FT12F001 DD SYSOUT=(*,5217),
           Duplex=No,
```

Options "FSP" was necessary to allow batch processing of proc fsedit, and OPTCD=J referenced the print control language in the body of the letter.

### Fonts

Fonts were added to a table through a batch job which copied the binary stream of bits to disk. We currently have ten fonts in the table. These are accessed by number in the letter.

## SUMMARY AND FUTURE APPLICATIONS

Through the use of SAS Macros, SAS datasets, the FSLETTER procedure, and our Xerox printer, we have been able to produce a quality product for our advertising department. The cost and time savings alone have made this a much sought after system. We plan to execute letters not only from our customer database, but from our prospect files and the databases of the company's divisions and subsidiaries. We also have plans to experiment with PC SAS, when it becomes available to us, so we can locally print on any letterhead, print envelopes, and insert cards for personalized mailings. The use of this product has, in some ways, revolutionized the way we use our customer database. We will continue to explore new ways to incorporate FSLETTER in many of our current database related operations.

## APPENDIX A

```
//E21SPIR JOB (BSD,0208,,1,999), 'MOLMAR BOX002', CLASS=B,
//MSGCLASS=X, NOTIFY=R05A03
//NORUM OUTPUT JDE=610051
//MACLET EXEC SAS, OPTIONS='FSP,9=72,MACRO', REGION=6000K
//WORK DD SPACE=(CYL,(100,70))
//FT12F001 DD SYSOUT=(*,5217), DUPLEX=NO,
//FT12F001 DD SYSOUT=*, DUPLEX=NO,
// DCB=(BLKSIZE=141, LRECL=137, RECFM=VBA, OPTCD=J),
// OUTPUT=* NORUM
//SASLIB DD DSN=SAS.FORMAT.TXTLIB, DISP=SHR
//SPIRDATA DD DSN=TSSTPLS.SPIRS.DBASE.UNLOAD.FILE(0),
// DISP=OLD
//SAS6 DD DSN=SAS.SAS6.SASDATA, DISP=OLD
//TEMP DD DSN=LPHETWP.UNIT=SYSDA, EFBC=(CYL,(100,10))
//MACLIB DD DSN=SAS.MC.MACROS.CMT2, DISP=SHR
//OPTIONS NOSOURCE MACROGEN MPRINT ;
COMMENT *****
** PGM: MACLETFR MACRO CALL FOR FREE TRIAL LETTERS **
** SPIRLET-BRINGS IN SEGMENT DATA FROM UPLOAD SPIRS **
** HLETIN- BRINGS HEADER DATA FROM UPLOAD SPIRS **
** ADVLET3-SELECTS BASED ON DATE LETTERS TO BE **
** PRINTED. THIS CALLS MCASE WHICH MIXES CASE **
** NAME, TITLE, COMPANY-ALSO CREATES SALUTATION **
*****
OPTIONS SOURCE ;
OPTIONS DQUOTE;
OPTIONS USER=TEMP;
%INC MACLIB(SPIRLET); COMMENT USE FOR SPIRLET OR HLETIN ;
%INC MACLIB(MCASE ADVLET3);
COMMENT USE IF RUNNING ADVLET3 ONLY;
COMMENT DON'T FORGET TO CHANGE JCL FOR PAPER!!!!!!;
%SPIRLET(DAYS=60);
%HLETIN(DAYS=60);
*SENDLET(PCODE='CPSX183', LETCODE=CPSX183);
*SENDLET(PCODE='CPSX184', LETCODE=CPSX183);
*SENDLET(PCODE='CPSX483', LETCODE=CPSX183);
*SENDLET(PCODE='ITRX182', LETCODE=ITRX182);
*SENDLET(PCODE='ITRX183', LETCODE=ITRX183);
*SENDLET(PCODE='PSLR181', LETCODE=PSLR181);
*SENDLET(PCODE='PSLR481', LETCODE=PSLR181);
*SENDLET(PCODE='PSLR482', LETCODE=PSLR181);
*SENDLET(PCODE='PSLR191', LETCODE=PSLR191);
*SENDLET(PCODE='PSLR491', LETCODE=PSLR191);
RUN;

%MACRO SPIRLET(DAYS=);
*****
** PGM:SPIRLET PDS: SAS.MC.MACROS.CMT2 **
** COMMENT ....SPIRLET--PROGRAM TO BRING IN SPIRS RAW DATA **
** AND MAKE A SAS DATASET OF UNIQUE MATCH CODE AND PRODUCT**
** COMBINATIONS. **
*****
DATA SPIRSEGS;
INFILE SPIRDATA;
RETAIN MATCH CD;
TDAY=DATE();
FORMAT TDAY MMDDYY6.;
TDAYJUL=JULDATE(TDAY);
INPUT MATCH CD $ 1-16 @294 SCOUNT IB2. @;
M=296;
DO I = 1 TO SCOUNT;
INPUT @N +8 STATUS $ 1. @N +9 STARTD PD4.
@N +23 PROMCODE $9. @N +39 ACCT_OR $1. @;
IF STATUS = 'A' AND (ACCT_OR = 'A' OR ACCT_OR = 'T') AND
(SUBSTR(PROMCODE,1,7)='CPSX183'
OR SUBSTR(PROMCODE,1,7)='CPSX184'
OR SUBSTR(PROMCODE,1,7)='CPSX483'
OR SUBSTR(PROMCODE,1,7)='PSLR181'
OR SUBSTR(PROMCODE,1,7)='PSLR481'
OR SUBSTR(PROMCODE,1,7)='PSLR482'
OR SUBSTR(PROMCODE,1,7)='PSLR191'
OR SUBSTR(PROMCODE,1,7)='PSLR491'
OR SUBSTR(PROMCODE,1,7)='ITRX182'
OR SUBSTR(PROMCODE,1,7)='ITRX183') THEN DO;
INPUT @N PRODUCT $8.
@N +21 REASON $2.
@N +32 ACCT_NO $7.
@N +44 NUM_COPY 4. @;
M=N + 50;
IF STATUS = 'A' AND (ACCT_OR = 'A' OR ACCT_OR = 'T') AND
(SUBSTR(PROMCODE,1,7)='CPSX183'
OR SUBSTR(PROMCODE,1,7)='CPSX184'
OR SUBSTR(PROMCODE,1,7)='CPSX483'
OR SUBSTR(PROMCODE,1,7)='PSLR181'
OR SUBSTR(PROMCODE,1,7)='PSLR481'
OR SUBSTR(PROMCODE,1,7)='PSLR482'
OR SUBSTR(PROMCODE,1,7)='PSLR191'
OR SUBSTR(PROMCODE,1,7)='PSLR491'
OR SUBSTR(PROMCODE,1,7)='ITRX182'
OR SUBSTR(PROMCODE,1,7)='ITRX183') THEN OUTPUT;
END;
END;
DATA SPIRSEGS; SET SPIRSEGS;
DROP SCOUNT I STARTD START2 N;
START2=INPUT(PUT(STARTD,6.),MMDDYY6.);
START B=START2;
VAR=SUBSTR(PROMCODE,5);
PROC SORT; BY MATCH_CD;

DATA SAS6.SPIRLET;SET SPIRSEGS;
BY MATCH_CD;
%MEMD SPIRLET;
```

```

MACRO MLETIN(DAYS=);
*****
PGMS: MLETIN          POS: SAS.MC.MACROG.CONT2 *;
** COMMENT...SPIRHEAD--PROGRAM TO BRING IN SPIRS RAW DATA*
** AND FORM A SAS DATASET OF THE HEADER INFORMATION ONLY **
*****

DATA MASTSPIR;
INFILE SFIRDATA;
INPUT MATCH_CD $ 1-16 ZIP $ 2-6 SP_NAME $ 17-44 DISTRICT
      $ 45-47 PCR_CD $ 51-51 SEX_CD $ 59-59 SIC $ 68-71
      SP_TITLE $ 85-112 TITLE_D $ 72-76 TITLE_M $ 77-81
      SP_COMP $ 114-441 DIVISION $ 143-170 STREET $ 172-199
      CITY $ 201-217 STATE $ 218-219 TELE $ 221-230;
PROC SORT; BY MATCH_CD;

DATA SPIRLET; MERGE MASTSPIR (IN=A) SAS6.SPIRLETT (IN=B)
              BY MATCH_CD;
              IF A AND B THEN OUTPUT SPIRLET;
              IF A AND C THEN OUTPUT FLXDATA;
PROC SORT DATA=SPIRLET;
BY MATCH_CD ACCT_NO;

DATA SPIRMAST; SET SAS6.SPIRLET;
PROC SORT;
BY MATCH_CD ACCT_NO;

DATA DUPS NEWTRL; MERGE SPIRMAST(IN=A) SPIRLET(IN=B);
                  BY MATCH_CD ACCT_NO;
                  IF A AND B THEN OUTPUT DUPS;
                  IF B AND ^A THEN OUTPUT NEWTRL;
PROC PRINT DATA=NEWTRL;
VAR SP_COMP SP_NAME CITY STATE PROMCODE START_D;
TITLE 'NEW TRIALS FROM SPIRS';
DATA NEWTRL; SET NEWTRL;
  SP_NAME2=SP_NAME;

DATA SAS6.SPIRLET; SET SAS6.SPIRLET NEWTRL;
DROP LAST FIRST A;
  A='<';
IF INDEX(SP_NAME,A)=0 THEN DO;
  SP_NAME='>';
END;
ELSE IF INDEX(SP_NAME,A) GT 0 THEN DO;
  LAST=SCAN(SP_NAME,1,'<');
  FIRST=SCAN(SP_NAME,2,'<');
  FIRST=TRANSLATE(FIRST,' ','');
  SP_NAME=TRIM(FIRST)||'>'||(LAST);
  SP_NAME=TRANSLATE(SP_NAME,' ','$');
END;

%MEND MLETIN;

MACRO SENDLET(PCODE=,LETCODE=);
*****
PGMS: ADVLET3        POS: SAS.MC.MACRO5.CONT2 *;
** COMMENT...MACRO TO SEND LETTER TO 45-DAY APPROVAL SUBS *
** FOR ADVERTISING PCODE=PROMOTION CODE LETCODE=LETTER TO *
** BE MAILED **
*****
DATA SENDLET OVERDUE; SET SAS6.SPIRLET;
IF SUBSTR(PROMCODE,1,7)=-4PCODE;
  TODAY=DATE();
  FORMAT TODAY MMDDYY6.;
  TODAYJUL=JULDATE(TODAY);
  SDATEJUL=JULDATE(START_D);
  KEY=SUBSTR(PROMCODE,1,7);
  SENDDATE= TODAYJUL-SDATEJUL;
IF SENDDATE GE 35 AND SENDDATE LE 45 THEN OUTPUT SENDLET;
IF SENDDATE GT 45 THEN OUTPUT OVERDUE;
PROC SORT DATA=OVERDUE;
BY MATCH_CD ACCT_NO KEY;

PROC SORT DATA=SENDLET;
BY MATCH_CD ACCT_NO KEY;
DATA OVERDUE; MERGE OVERDUE(IN=A) SAS6.SENDLET(IN=B);
              BY MATCH_CD ACCT_NO KEY;
              IF A AND ^B;
PROC PRINT DATA=OVERDUE;
DATA SENDLET; MERGE SENDLET(IN=C) SAS6.SENDLET(IN=D);
              BY MATCH_CD ACCT_NO KEY;
              IF C AND ^D;
DATA SENDLET; SET SENDLET OVERDUE;
SENDDATE=TODAY();
$UPLOW;
PROC PSEEDIT DATA=SENDLET
  LETTER=SAS6.LETTER DDPRINT=FT12F001
SEND=&LETCODE;
PROC FORMS I = 1
          S = 2
          D = 16
          H = 4
          W = 30
          L = 7
          NA = 2;
LINE 1 PROMCODE/R;
LINE 2 SURNAME/R;
LINE 3 TITLE/R;
LINE 4 COMPALL/R;
LINE 5 DIVISION/R;
LINE 6 STREET/R;
LINE 7 CTSTEE/P R;
TITLE 'NAMES AND ADDRESSES FOR &PCODE';

```

```

DATA SENDLET; SET SENDLET;
KEEP ACCT_NO KEY MATCH_CD SENDDATE;
KEY=SUBSTR(PROMCODE,1,7);
PROC SORT; BY MATCH_CD ACCT_NO KEY;

DATA SAS6.SENDLET; MERGE SENDLET SAS6.SENDLET;
BY MATCH_CD ACCT_NO KEY;
RUN;
%MEND SENDLET;

```

#### APPENDIX B

```

MACRO UPLOW;
OPTIONS MOCAPS;
COMMENT *****
* PROGRAM: MCASE DSN: SAS.MC.MACRO5.CONT2 *
* ALGORITHM TO CONVERT FROM UPPER CASE TO MIXED CASE *
*****
* CONVERTS FIRM TO UPPER/LOWER CASE;

LENGTH COMPALL $60;

WORD1 = SCAN(SP_COMP,1,' ');
WORD2 = SCAN(SP_COMP,2,' ');
WORD3 = SCAN(SP_COMP,3,' ');
WORD4 = SCAN(SP_COMP,4,' ');
WORD5 = SCAN(SP_COMP,5,' ');

PART1A = SUBSTR(WORD1,1,1);
PART1B = SUBSTR(WORD1,2);
PARTW = TRANSLATE(PART1B,'abcdefghijklmnopqrstuvwxyz,-&','
  'ABCDEFGHIJKLMNQPQRSTUVWXYZ,-&');

PART2A = SUBSTR(WORD2,1,1);
PART2B = SUBSTR(WORD2,2);
PARTX = TRANSLATE(PART2B,'abcdefghijklmnopqrstuvwxyz,-&','
  'ABCDEFGHIJKLMNQPQRSTUVWXYZ,-&');

PART3A = SUBSTR(WORD3,1,1);
PART3B = SUBSTR(WORD3,2);
PARTY = TRANSLATE(PART3B,'abcdefghijklmnopqrstuvwxyz,-&','
  'ABCDEFGHIJKLMNQPQRSTUVWXYZ,-&');

IF WORD4 NE ' ' THEN DO;
PART4A = SUBSTR(WORD4,1,1);
PART4B = SUBSTR(WORD4,2);
PARTZ = TRANSLATE(PART4B,'abcdefghijklmnopqrstuvwxyz,-&','
  'ABCDEFGHIJKLMNQPQRSTUVWXYZ,-&');
END;

IF WORD5 NE ' ' THEN DO;
PART5A = SUBSTR(WORD5,1,1);
PART5B = SUBSTR(WORD5,2);
PARTQ = TRANSLATE(PART5B,'abcdefghijklmnopqrstuvwxyz,-&','
  'ABCDEFGHIJKLMNQPQRSTUVWXYZ,-&');
END;

COMPALL = TRIM(PART1A)||TRIM(PARTW)||' '||TRIM(PART2A)||
  TRIM(PARTX)||' '||TRIM(PART3A)||TRIM(PARTY)||' '||
  TRIM(PART4A)||TRIM(PARTZ)||' '||TRIM(PART5A)||PARTQ;

COMPALL = TRIM(LEFT(COMPALL));

```



APPENDIX D  
FSLETTER attribute screen

```
Command ==>
Symbolic name: SUBNAME      Occurrence: 1   Line: 4   Column: 4

Value attributes:  _ Lastname  _ Right justify  _ Flow after  X Delete
                   _ Dear      _ Required      _ Flow before

Field color: B   G=Green B=Blue R=Red P=Pink C=Cyan Y=Yellow W=White

Field attr: H   H=Highlight R=Reverse video B=Blink U=Underline

SAS Variable aliases: SUBNAME .....
```

APPENDIX E  
Text Body and Margin Information

```
Command ==>

Text Body Information
Characters per Line: 81   Lines on First Page: 63   Lines on Following: 63

First Page Margins   Left:  _   Top: 3   Bottom: 3
Following Pages     Left:  _   Top: 3   Bottom: 3
```

TSO Print File Parameters

```
Command ==>
Destination:  SALPRT1      Class: A      Program: _____
Forms:        FSLF        Ucs:  _____  Outlin: _____
Copies:       1           Fcb:  _____  Hold:  _____

Carriage Control Information  No CC: X

Signal Page Skips Before      First Print Control Language:  X
                              First Text Page:  X
                              Following Print Control Language: X
                              Following Text Pages:  X

Additional information needed only for IBM 3800 printers
Character Tables:  _____  Burst:  _____  Optcode=J:  X
Flash Name:  _____  Flash Count:  _____  Modify Name:  _____  Modify Trc:  _____
```

APPENDIX F



THE BUREAU OF NATIONAL AFFAIRS, INC.

March 9, 1988

William C. Knapp  
425 Provident Bank Bldg  
7th & Vine Street  
Cincinnati, OH 45202

Dear Subscriber:

Thank you for letting us share five weeks of International Trade Reporter's CURRENT REPORTS with you.

During your trial period, CURRENT REPORTS has kept you up to date on congressional and White House trade policy activity. You've monitored U.S. and Canadian efforts on the U.S.-Canada free trade agreement . . . international reaction to the new U.S. trade law . . . latest U.S. retaliatory actions . . . and a host of other important trade issues.

You've read our objective news reports and legislative analyses written especially for your information needs.

Now I invite you to continue receiving this unrivaled notification and analysis by subscribing to International Trade Reporter's CURRENT REPORTS.

Accept this invitation and let CURRENT REPORTS continue to keep you informed and prepared for tomorrow's tough decisions. A one-year subscription is only \$668 -- an excellent value for a full year's issues packed with the information you need to stay on top of latest developments, changing restrictions . . . and new opportunities.

I've enclosed a postpaid Invitation Acceptance Card for your convenience -- simply complete it and drop it in the mail today and we'll begin your subscription to International Trade Reporter's CURRENT REPORTS immediately.

Sincerely,

Robert K. Fuchs  
for the Editors

RKF:noc  
183A

P.S. A Toll-Free call to BNA's Response Center will speed your order:  
1-800-372-1033.

1231 Twenty-fifth Street, Northwest, Washington, DC 20037 □ Telephone (202) 452-4200 □ TELEX: 285656 BNAI WSH

*For further information, contact:*  
Phyllis Jordan-Johnson at (202) 452-4215 or  
Margie Molnar at (202) 728-3388

©SAS, SAS/FSP are registered trademarks of SAS  
Institute, Inc., Cary, NC, USA