Character Based / Object Oriented Reporting with Procedure and PUT Style Output. Annotate Character Based Reports!

M.P. Welch
Sprint Corporation

Abstract

This paper describes a technique which allows placement of procedure output and freeform (PUT style) text anywhere on any page of a SAS character based report. The technique can be used under all SAS versions and supported operating systems. The example code is from an MVS operating system running SAS 6.08.

Introduction

Today’s world of GUI interfaces and graphically based output is increasingly replacing character based reporting. However, many still find themselves in environments where character based reports are used extensively. Character based reports may not be sexy but they still have the advantage of being easily printed, stored, copied, faxed, included in emails, and exchanged between computer systems.

The technique described in this paper allows the programmer to use each page of printed output as a blank canvas on which SAS generated character output may be easily placed.

When creating SAS character reports it seems the SAS programmer is always faced with the trade-off of using canned procedure based output or using tedious and time consuming PUT style output. This paper allows both techniques to be used together to create unique reporting possibilities.

Process Flow

This technique involves three phases. The first phase creates the output objects using procedure and put style output. These outputs are written to a temporary file. In the second phase, the temporary file of outputs is processed and each record of each output is assigned page, row, and column attributes based on the first title line of procedure output or first put statement of PUT style output. The third phase simply outputs these records.

A temporary JCL or libname allocation is required to hold the procedure output and freeform text output before the final output. FT20F001 was chosen as the DD name to be consistent with previous versions of SAS.

Two %INCLUDE statements are used to logically separate the object creation process from the placement process. OPTIONS NOCENTER is required for proper formatting of final output.

```sas
/*S1 EXEC SAS */
/*FDS DD DISP=SHR */
/*FT20F001 DD DISP=SHR */
/*SYSIN DD OPTIONS NOCENTER; */
/*INCLUDE FDS(CHAR); */
/*INCLUDE FDS(PUTCNT); */
```
Creating Outputs

Outputs can be SAS procedure output and/or PUT style output. In this example PROC PRINT, PROC FREQ, PROC TABULATE, and PROC PLOT output is combined with PUT style output on the same page. A simple sample data set of three observations each with three variables is used for all procedures.

```
DATA SAMPLE;
  INPUT A B C;
  CARDS:
  1 2 3
  4 5 6
  7 8 9 ;
```

Each procedure is enclosed within a PROC PRINTTO which redirects the output to the temporary file. Placement information is specified in the first TITLE line of each procedure. This title line is ‘borrowed’ to allow each procedure’s output to be placed by specifying the location of the leftmost position of the first line of the output. Each subsequent line of output of the procedure is calculated based on the title’s page, row, column specification.

```
PROC PRINTTO UNIT=20;
  PROC PRINT DATA=sample;
  TITLE1 \$PAGEROWCOL(001, 033, 50) ;
  TITLE2 \$PROC PRINT OUTPUT ;
  TITLE3 \$-------------- ;
  PROC PRINTTO ;
```

```
PROC PRINTTO UNIT=20;
  PROC FREQ DATA=sample;
  TITLE1 \$PAGEROWCOL(001, 027, 3) ;
  TITLE2 \$PROC FREQ OUTPUT ;
  TITLE3 \$-------------- ;
  PROC PRINTTO ;
```

```
PROC PRINTTO UNIT=20;
  PROC PLOT DATA=sample NOLEGEND;
  TITLE1 \$PAGEROWCOL(001, 001, 001) ;
  TITLE2 \$PROC PLOT OUTPUT ;
  TITLE3 \$-------------- ;
  PLOT A=B=C / BOX HPOS=24 VPOS=12 ;
  PROC PRINTTO ;
```

PUT style output is added to the temporary file with the FILE statement and MOD parameter. The first line of PUT output specifies the page, row, and column where the put output will be placed in the final report.

```
DATA _NULL_;
  FILE PT2F001 PRINT N=PS P=500 LS=140
    LINESLEFT=0 NOTITLES MOD;
  PUT 801 \$PAGEROWCOL(001, 046, 054) ;
  PUT 801 + ------------------------------ ;
  PUT 801 + QUESTIONS/COMMENTS: + ;
  PUT 801 + ;
  PUT 801 + KEWKILCE@COMPOSERV.COM + ;
  PUT 801 + +------------------------------- ;
```

```
DATA _NULL_;
  FILE PT2F001 PRINT N=PS P=500 LS=140
    LINESLEFT=0 NOTITLES MOD;
  PUT 801 \$PAGEROWCOL(001, 019, 033) ;
  PUT 801 + +------------------------------- ;
  PUT 801 + THIS REPORT IS CREATED WITH A COMBINATION + ;
  PUT 801 + OF PROCEDURE AND PUT STYLE OUTPUT, ABOUT 30 + ;
  PUT 801 + LINES OF CODE PLACES ALL OUTPUT OBJECTS + ;
  PUT 801 + BASED ON THE PAGE, ROW, AND COLUMN + ;
  PUT 801 + SPECIFIED IN THE FIRST TITLE LINE OF THE + ;
  PUT 801 + PROCEDURE OUTPUT OR THE FIRST PUT STATEMENT + ;
  PUT 801 + IN THE CASE OF PUT STYLE OUTPUT. + ;
  PUT 801 + +------------------------------- ;
```

After the PROC and PUT outputs are written to the temporary file, the format program then reads the temp file and creates a SAS data set with four variables. The four variables are PAGE, ROW, COL, and RECORD. PAGE, ROW, and COL are reset when the $PAGEROWCOL string is found in the temporary data set otherwise ROW is incremented by one. The RECORD variable contains a record of the temporary file.
An adjustment to column is automatically performed when procedure output does not start in column 1. After conversion to a SAS data set, the outputs are simply printed with PUT #ROW @ADJ.

/* SNOWFALL */

CONCLUSION

The technique described above is based on 30 lines of code which can be altered to include any type of post processing of procedure output. Procedure output created with BY groupings could be processed by multiple passes with WHERE statements or by processing $PAGEROWCOL from one of the BY lines. Adding PUT comments to full page plots based on data values, creating macro put objects, overprinting for emphasis or overlaying unwanted portions of output with blanks is also possible.

If you're creating character based reports and want the power of procedure output with the customizability of put style output this technique may be for you.
SAMPLE REPORT INCLUDES PROC PLOT, PROC TABULATE, PROC PRINT, PROC FREQ, AND PUT STYLE OUTPUT.

PROC PLOT OUTPUT
-------------------
A 1 +
 7 +
 1 +
 1 +
 4 +
 1 +
 1 +
 1 +
 1 +
 1 +
 1 +
 1 +
 2 +
 5 +
 8 +

PROC TABULATE OUTPUT
-----------------------
[SAMPLE ] [ B ] [ C ]
[------- ] [ ] [ ]
[A ] [ ] [ ]
[ ] [ ] [ ]
[ ] [ ] [ ]
[ ] [ ] [ ]
[ ] [ ] [ ]
[ ] [ ] [ ]
[ ] [ ] [ ]
[ ] [ ] [ ]
[ALL ] [ ] [ ]

PROC FREQ OUTPUT
-------------------
A Frequency Percent
-------- --------
1 1 33.3
4 1 33.3
7 1 33.3

Cumulative Frequency Percent
---------------------------------------
1 1 33.3
2 2 66.7
3 3 100.0

B Frequency Percent
-------- --------
2 1 33.3
5 1 33.3
8 1 33.3

Cumulative Frequency Percent
---------------------------------------
2 1 33.3
2 2 66.7
3 3 100.0

C Frequency Percent
-------- --------
3 1 33.3
6 1 33.3
9 1 33.3

Cumulative Frequency Percent
---------------------------------------
3 1 33.3
6 2 66.7
9 3 100.0

PROC PRINT OUTPUT
-------------------
OBS A B C
1 1 2 3
2 4 5 6
3 7 8 9

QUESTIONS/COMMENTS:
MPWELCH@CONFUSEYOU.COM