Utilizing Clinical SAS Report Templates

Sunil Kumar Gupta
Gupta Programming, Thousand Oaks, CA

ABSTRACT

SAS programmers often have the responsibility of supporting the reporting needs of the Clinical Affairs Department of Pharmaceutical Companies. This involves generating a variety of reports to fulfill the regulatory requirements of the clinical investigation of the medical device. By establishing clinical reporting templates for each of the functional aspects of Clinical Data Management, the SAS programmer can dramatically improve the efficiency of developing and generating reports.

Examples of these functional reports include patient listings, patient x-ray logs, monitor site visits, monthly status reports, & reports necessary for pre market approval application. Each type of report requires different layouts due to the different objectives of the organization. By establishing a reporting template for each type, the SAS programmer can make modifications to an already verified and complete program. Only minor adjustments may be required for user specific selections.

Proper tools such as SAS report templates should be in place to facilitate the rapid development and execution of clinical reports. Establishing reporting standards will increase efficiency.

INTRODUCTION

Supporting the reporting needs of the Clinical Affairs Department can be a challenging task. This is because of the pressure on the programmer to quickly generate a variety of reports. Time is often critical as millions of dollars in sales can be lost due to delays in submission.

By establishing clinical reporting templates for each of the functional aspects, the SAS programmer becomes organized and prepared to receive reporting requests. The department improves their efficiency in the development and generation of clinical reports.

System standardization can be realized in the following areas:

- Program Header
- File Definitions
- Merged File Definitions
- Field Properties
- Standard & Custom Conditions
- Custom Formats
- Random Sample
- Verification of Dataset Contents
- Output Layout

If possible, it is recommended to use the report specification similar to one defined in the SAS Service Request Form before initiating the coding.

This paper will review sample templates for detail & summary patient listings.

SYSTEM STANDARDIZATION

Program Header

By completing the program header, proper documentation is maintained for all programs. This is essential for program support.

File Definitions

All files containing the clinical information are defined with any access & view descriptors. This is a central source of file documentation.

Merged File Definitions

All links are defined and established to create new files for data consolidation & reporting.

Field Properties

For each file, fields are defined with label, format, & informat statements. These properties are utilized in all data analysis and presentation.

Standard & Custom Conditions on the file

Any required conditions can be applied to assure data validity. Any user specific conditions can also be applied.

Custom Formats

Standards in study formats can be recognized.

Random Sample

With a small random sample, a sufficient amount of data is available for testing. This allows for rapid testing without utilizing all the data in the dataset.

Verification of Dataset Contents

This is important to assure proper data extraction and to verify programming assumptions.

Output Layout

The end user knows what to expect on the report. The header contains the title, date of execution, and selection conditions.

The following procedures are presented with the macro language: DATA Step, PROC SORT, FORMAT & REPORT.

SAMPLE DATASETS

The two datasets represent patient demographics and follow-up visit information.

```
* Program Header;
* Name              - List1;
* Function          - Detail & Summary listing of Clinical
Data;
* Programmer        - Sunil K. Gupta;
* Date              - 7/1/95;
* Verification Date - 7/2/95   By - ;
* Revision Log;
* Date     Programmer                    Change;

LIBNAME CLINRPT 'C:\SAS\CLINICAL\MONTHLY';

* Patient Demographics File;

DATA DEMO;
INPUT INV 1-2 PATNO 4-5 PTNAME $5.
      STATUS 12 SEX 14 AGE 16-18 DX 20;

        * Custom Conditions;
        * WHERE INV IN (1,2)
                AND STATUS IN (1)
                AND SEX IN (1)
                AND AGE < 45
                AND DX IN (1);

        * Random Sample;
        * IF RANUNI(-1) LE 0.1;

        * Field Properties;
        LABELS
        INV             = 'Investigator'
        PATNO           = 'Patient*#'
        PTNAME          = 'Patient*Name'
        STATUS          = 'Status'
        SEX             = 'Sex'
        AGE             = 'Age'
        DX              = 'Diagnosis';

        FORMAT
        INV             2.
        PATNO           2.
        PTNAME          $5.
        STATUS          1.
        SEX             1.
        AGE             3.
        DX              1.;

        CARDS;
1   1  JOHN   1  1  31    2
1   2  JAN    2  1  41    1
2   1  TIM    1  2  100   1
3   1  JOE    1  2  32    3
4   1  SUE    2  2  42    3
5   1  MARY   1  2  102   3
;
RUN;
```

```
* Patient Followup Visits File;

DATA PVISIT;
INPUT INV 1-2 PATNO 4-5 VISIT 7-8 EFF1 10-11
      EFF2 13-14 SAF1 16-17;

        * Custom Conditions;
        * WHERE INV IN (1,2)
                AND VISIT IN (1,2,3);

        * Random Sample;
        * IF RANUNI(-1) LE 0.1;

        * Field Properties;
        LABELS
        INV             = 'Investigator'
        PATNO           = 'Patient*#'
        VISIT           = 'Visit'
        EFF1            = 'Efficacy*Variable*1'
        EFF2            = 'Efficacy*Variable*2'
        SAF1            = 'Safety*Variable*1';

        FORMAT INV          2.
               PATNO        2.
               VISIT        1.
               EFF1         2.
               EFF2         2.
               SAF1         2.;
        CARDS;
1   1  1   50 52 62
1   1  2   60 41 73
1   1  3   70 12 25
1   1  4   80 41 75
1   1  5   90 25 72
1   2  1   40 13 36
1   2  2   50 15 26
1   2  3   60 18 42
2   1  1   80 17 53
2   1  2   90 26 47
;
RUN;

* Standard Study Format;
PROC FORMAT;
        VALUE INV
        1 = 'Investigator 1'
        2 = 'Investigator 2'
        3 = 'Investigator 3'
        4 = 'Investigator 4'
        5 = 'Investigator 5';

        VALUE STATUS
        1 = 'Active'
        2 = 'Dead';

        VALUE SEX
        1 = 'Male'
        2 = 'Female';
```

```
        VALUE DX
        1 = 'OA'
        2 = 'RA'
        3 = 'PT';

        VALUE VISIT
        1 = 'Preop'
        2 = 'Operative'
        3 = 'Week 1'
        4 = 'Week 2'
        5 = 'Week 3';
        QUIT;

* Merged File - Demographics & Followup File;

PROC SORT DATA=DEMO;
        BY INV PATNO;
        RUN;

PROC SORT DATA=PVIST;
        BY INV PATNO;
        RUN;

DATA PDVISIT;
        MERGE DEMO (IN=A) PVISIT (IN=B);
        BY INV PATNO;
        * IF RANUNI(-1) LE 0.1;
        IF A;
        RUN;


* Verification of Dataset Contents;
PROC FREQ DATA=PDVISIT;
        TABLES INV PATNO STATUS SEX DX VISIT;
        RUN;
```

MACRO VARIABLES

```
%LET DATAV     = PDVISIT;
%LET CLASSV    = INV PATNO VISIT;
%LET VARV      = EFF1;
%LET STATV1    = MEAN;
```

Level 1 Controls

Macro variables enable the programmer to specify the data set name, factor & analysis variable, & statistics. The factor variable may be more than one variable. The analysis variable is usually one of the efficacy or safety variables. The statistics option includes the following: mean, sum, min, max, & std.

```
%LET INVV      = (1, 2);
%LET PATNOV    = (1, 2, 3, 4, 5);
%LET VISITV    = (1, 2, 3);
```

Level 2 Controls

This enables the programmer to futher subset the data for additional analysis. For example, the programmer can specify the investigator number, patient number, or visit period. This section can be uncommeted if needed.

SIMPLE PATIENT LISTING

For a simple patient listing, the report procedure can be utilized.

Once the file is sorted, then PROC REPORT can be applied. This generates a sorted listing of the investigator name, patient number, visit period, and score value. See output 1 for the report generated.

```
PROC SORT DATA=PDVIST;
        BY INV PATNO;
        RUN;
TITLE1 "CLINICAL REPORT : Simple Patient Listing -
&SYSDATE, &SYSTIME";

PROC REPORT
DATA = &DATAV
OUTREPT = CLINRPT.LIST1
CENTER HEADLINE HEADSKIP MISSING LIST SPLIT='*';

        * Level 2 controls;
        * WHERE  INV IN &INVV
        AND PATNO IN &PATNOV
        AND VISIT IN &VISITV;

        * Apply custom formats;
        FORMAT INV INV. VISIT VISIT.;

        * List column variables;
        COLUMN &CLASSV &VARV;

        * Define each column field;
        DEFINE INV / GROUP CENTER
        'Investigator*Name';

        DEFINE PATNO / GROUP CENTER  'Patient*#';

        DEFINE VISIT / GROUP CENTER     'Visit';

        DEFINE &VARV / &STATV1 CENTER
        'Efficacy*Variable*1'
        FORMAT=COMMA6.;


        * Generate a line after each Investigator;
        COMPUTE AFTER INV;
          LINE 45*'-';
          LINE 'Efficacy Variable 1 Scores for ' INV 2.
                +2 ' Investigator  ';
          LINE ' ';
        ENDCOMP;
        RUN;
```

OUTPUT 1

CLINICAL REPORT : Simple Patient Listing - Monday, 3JULY95 15:03

INVESTIGATOR NAME	PATNO #	VISIT	EFFICACY VARIABLE 1
Investigator 1	1	Operative	60
		Preop	50
		Week 1	70
		Week 2	80
		Week 3	90
	2	Operative	50
		Preop	40
		Week 1	60
Efficacy Variable 1 Scores for 1 Investigator			
Investigator 2	1	Operative	90
		Preop	80
Efficacy Variable 1 Scores for 2 Investigator			

SUMMARY PATIENT LISTING

For a summary listing across follow-up visits, the REPORT procedure is applied again. This procedure generates a summary listing by investigator name and patient number. The visit scores reported include: mean, minimum, maximum, and standard deviation. See output 2 for the report generated.

```
%LET CLASS=INV PATNO;

TITLE1 "CLINICAL REPORT : Summary Patient Listing -
&SYSDATE, &SYSTIME";

PROC REPORT
DATA = &DATAV
OUTREPT = CLINRPT.LIST2
NOWD HEADLINE HEADSKIP MISSING LIST SPLIT='*';

        * Level 2 controls;
        * WHERE  INV IN &INVV
        AND PATNO IN &PATNOV
        AND VISIT IN &VISITV;

        * List column fields;
        COLUMN &CLASSV &VARV
        &VARV=SMIN &VARV=SMAX
        &VARV=SSTD;

        * Apply custom formats;
        FORMAT INV INV.;

        * Define each column fields;
        DEFINE INV / GROUP CENTER
        'Investigator*Name';

        DEFINE PATNO / GROUP CENTER  'Patient*#';

        DEFINE &VARV / &STATV1 CENTER
        'Efficacy*Variable*1'
        FORMAT=COMMA6.;

        DEFINE SMIN / MIN CENTER
        'MIN*SCORE' FORMAT=COMMA6.;

        DEFINE SMAX / MAX CENTER
        'MAX*SCORE' FORMAT=COMMA6.;

        DEFINE SSTD / STD CENTER
        'STD*SCORE' FORMAT=COMMA6.;

        COMPUTE AFTER INV;
          LINE 45*'-';
          LINE 'Efficacy Variable 1 Scores for ' INV 2.
                +2 ' Investigator  ';
          LINE ' ';
        ENDCOMP;

        RUN;
```

OUTPUT 2

CLINICAL REPORT : Summary Patient Listing - Monday, 3JULY95 15:03

INVESTIGATOR NAME	PATNO	MEAN SCORE	MIN SCORE	MAX SCORE	STD SCORE
Invest. 1	1	70	50	90	16
	2	50	40	60	10
Efficacy Variable 1 Scores for 1 Investigator					
Invest. 2	1	85	80	90	7
Efficacy Variable 1 Scores for 2 Investigator					

Use the REPORT = option for using a predefined clinical template for reporting of a data subset.

```
* Subset Dataset for Male patients;

DATA PDVISITM;
        SET PDVISIT;
        WHERE SEX = 1;
RUN;

%LET DATAV = PDVISITM;

PROC REPORT
DATA = &DATAV
REPORT = CLINRPT.LIST2;
RUN;
```

SUMMARY

A similar template can be developed for each of the other functional reports.

The two PROC REPORT OPTIONS - OUTREPT & REPORT allow for additional reporting power.

Use the OUTREPT = option to save the report template. This template can be utilized for other datasets of the same dataset structure. A subset population of the original dataset is an application example.

Use the REPORT= option to load the template containing the report definition to format the current SAS dataset.

PROC FORMS can be used to generate reports that resemble the case report form. This can be useful in assisting in the verification of the clinical data.

With the clinical report templates, efficiency is realized because the programmer starts working with code from a similar program. Often, there are similar studies being conducted and the method of clinical review is similar across the efficacy variables. In addition, a great deal of time is saved because the modified code does not need to go through a complete and comprehensive test as the original code has.

By effectively utilizing SAS's flexibility and power for report generation, it is possible to expedite the process of report development. Clinical SAS report templates serve as a set of tools to facilitate the rapid development and execution of clinical reports. The standards established also improve the communication and training of new SAS programmers.

REFERENCES

SAS®Procedures Guide, ver. 6, third edition; SAS® Guide to the Report Procedure - Usage and Reference, ver. 6, first edition;

TRADEMARK INFORMATION

ABOUT THE AUTHOR

The author welcomes your comments & suggestions.

Sunil Kumar Gupta
Gupta Programming
926 Tupelo Wood Ct.
Thousand Oaks, CA 91320-3650
(805) 499-9691
sunilgupta@aol.com

ACKNOWLEDGMENTS

The author would like to thank Dr. Fred Hoehler of Data Management Center, Santa Ana, CA and Kirk Paul Lafler of Software Intelligence Corporation, Spring Valley, CA for their invaluable assistance in the preparation of this paper.