

**A System Approach for the Storage and Retrieval
of Information in Clinical Trials**

Bee-Lian Chen and Patricia A. Majcher
Hoechst-Roussel Pharmaceuticals, Inc.

Introduction

An efficient data structure to store report description information on clinical studies has been developed at Hoechst-Roussel Pharmaceuticals, Inc. Information such as headings, titles, and format specifications can be handled as a data dependent system. Such a structure has the following advantages: (a) the same information is stored only at one place (b) only the necessary pieces of information are stored (c) the system that accesses the data is data independent (d) the processing and the outcome of the system are data dependent (e) the descriptive information of the reports always matches the body of the report. Macros are provided describing the techniques that convert the information into program code using the SAS® function SYMPUT and retrieve the information in macro variables using the SAS® function SYMGET.

Description of the data structure

Project information of clinical trials is divided into seven major areas. The data for each area are stored in separate tables created and maintained by ISPF Dialog Management Services (ISPF/DMS). Table entries are retrieved through their keys. A key is made up of one or more table variables and uniquely identifies a table entry. An utility in ISPF/DMS is used to create SAS® data sets from the entries in the table to allow the SAS® programs to access the information. One data set is created for each table. The name of each data set is P00# where # corresponds to the table number. Detailed information on each data set follows.

P001: Project number and name information

Variable name	Data attribute	Key field	Description
PROJECT	N	Y	Project number
DNAME	C	N	Internal name
GNAME	C	N	Generic name
TNAME	C	N	Trade name

P002: Protocol label and trademark information

Variable name	Data attribute	Key field	Description
PROJECT	N	Y	Project number
PROTOCOL	N	Y	Protocol number
HFMT	N	N	Heading format
PROT	C	N	Alternative way to label the study
REGTMK	C	N	Flag for the registered trademark

P003: Investigator information.

Variable name	Data attribute	Key field	Description
PROJECT	N	Y	Project number
PROTOCOL	N	Y	Protocol number
INVEST	N	Y	Investigator number
INVLAST	C	N	Last name of the investigator

P004: Data record identification number

Variable name	Data attribute	Key field	Description
PROJKEY	C	Y	Project code of the study
ADMISS	N	N	Number of the admission record
DOSAGE	N	N	Number of the dosage record
OSPLASM	N	N	Number of the plasma record
OSURINE	N	N	Number of the urine record

P005: Specifications of the sample and assay

Variable name	Data attribute	Key field	Description
PROJKEY	C	Y	Project code
VARNUM	N	Y	Number assigned to a compound
DECML	N	N	Number of decimal places recorded
MEDIUM	C	N	The medium of the sample
SENSIT	N	N	The sensitivity of the assay
UNIT	C	N	The unit of the assay readings

P006: Secondary heading information

Variable name	Data attribute	Key field	Description
PROJKEY	C	Y	Project code
VARNUM	N	Y	Number assigned to a compound
TYPE	C	Y	The type of the text (H/L)
LINE	N	Y	The position of the text (1/2)
TEXT	C	N	Detail description of the content

PO07: "PROC FORMAT" information

Variable name	Data attribute	Key field	Description
PROJKEY	C	Y	Project code
FMT	N	Y	Number assigned to format group
FMINAM	C	Y	Name of the format
FMTVAL	N	Y	Value of the format
TEXT	C	N	Decode text for the format value

The following example is provided to describe the method of accessing the data set and the flexibility that the data sets contribute to the system. A brief description of each macro called in the example is given here; details of the macros are available in the appendix.

Macro %PARMSET sets up the parameters for a particular processing, declares the global macro variables from the information in data set PO05, generates the statements for the PROC FORMAT and executes them to provide the relevant decode information for a particular protocol. Macro %GETINFO uses MERGE to get the information on the heading from data sets PO01, PO02, and PO03 and concatenates the headings according to the format variable HFMT in data set PO02. Macro %FINDEX checks if the pattern INVESEI is part of the macro variable &PAGE and the macro is called several times within the macro %GETINFO. Macro %PUTINFO brings in the macro variables generated by macro %GETINFO to provide SAS® code for the headings of the report. Macro %FRMT generates the format according to decimal specification in data set PO05. Macro %PRINTV prints the data using the format generated in macro %FRMT.

Example

The following is a typical data set for bioavailability studies. Variables PROJECT and PROTOCOL identify the drug and the protocol number. VARNUM is the assigned number for a compound. GROUP identifies the treatment the subject received during the study. SUBJECT is the number assigned to each patient in the study. Y1 through Y6 are the assay readings for the six blood samples taken for each compound and subject.

The SAS® code for the example is given as follows:

```

%MACRO SAMPLE;
%LET PAGE=&XPAGE;
PROC SORT DATA=TEST;
BY VARNUM INVEST GROUP SUBJECT;
%GETINFO
DATA NULL;
SET TEST END=EOF;
BY VARNUM;
IF FIRST.VARNUM THEN DO;
  VN=1;
  DCODE='0'!!LEFT(VARNUM);
  LEN=7;
  DEC=SYNGET(DCODE);
  %FRMT
  CALL SYMPUT('F0'!!LEFT(VN),TRIM(LEFT(FRMT)));
END;
IF EOF THEN CALL SYMPUT('NVN',TRIM(LEFT(VN)));
DATA NULL;
SET TEST;
BY VARNUM INVEST GROUP SUBJECT;
FILE PRINT HEADER=NEWPAGE NOTITLES;
ARRAY TRACES(1) Y1-Y6;
IF FIRST.GROUP THEN PUT PAGE_;
IF FIRST.VARNUM THEN VN=1;
PUT @10 SUBJECT 4.0 @;
DO I=1 TO 6;
  COL=16-[(I-1)*9];
  %PRINTV
END;
PUT;
ITITLES;
RETURN;
NEWPAGE;
%PUTINFO
CPMNAME=TRIM(PUT(VARNUM,VARNUM));
PCODE='0'!!LEFT(VARNUM);
MEDLEN=SYNGET(PCODE);
UCODE='0'!!LEFT(VARNUM);
UNIT=SYNGET(UCODE);
PUT @18 'CONCENTRATION OF ' CPMNAME UNIT ' IN ' MEDIUM ///
@10 'TREATMENT: ' GROUP GROUP. //;
PUT @10 'SUBJ' 0 0.25 0.50 1.0 1.5 2.0'
OVERPRINT
@10 '
RETURN;
%MEND SAMPLE;
OPTIONS QUOTE MACROGEN SYMBOLGEN MLOGIC;
DATA TEST;
[INPUT PROJECT PROTOCOL VARNUM INVEST GROUP SUBJECT Y1 Y2 Y3 Y4 Y5 Y6;
CARDS;
999 104 1 23 10.08 2 3.05 3.09 2.94 2.88 3.10 3.02
999 104 1 23 10.08 3 3.67 3.80 3.83 3.83 3.57 3.67
999 104 1 23 10.06 4 1.22 1.14 1.09 1.07 1.10 1.23
999 104 1 23 10.12 2 2.89 2.98 2.87 3.02 3.38 3.71
999 104 1 23 10.12 3 4.78 4.78 4.52 4.96 5.19 5.49
999 104 1 23 10.12 4 2.04 1.60 1.58 1.60 1.78 2.10
999 104 1 23 20.08 1 4.20 4.10 4.08 4.42 4.68 5.03
999 104 1 23 20.08 5 3.88 3.92 3.88 3.66 3.70 4.07
999 104 1 23 20.08 6 4.32 4.48 4.39 4.53 4.83 4.55
999 104 1 23 20.12 1 3.64 3.87 4.24 5.16 5.00 5.03
999 104 1 23 20.12 5 2.62 3.41 4.06 5.06 5.53 5.22
999 104 1 23 20.12 8 5.08 4.99 5.03 5.02 4.84 5.04
999 104 3 45 10 2 0.0 0.1 34.8 52.2 103.1 189.1
999 104 3 45 10 3 28.2 26.9 30.7 215.1 319.1 484.1
999 104 3 45 10 4 11.5 11.3 13.0 26.0 96.8 139.1
999 104 3 45 20 1 15.5 12.4 12.8 10.4 13.2 18.5
999 104 3 45 20 5 12.7 9.2 9.9 18.6 21.9 40.5
999 104 3 45 20 6 13.5 15.1 11.2 48.0 72.1 145.9
RUN;
%PARMSET(PICKVAR=1 3,
PAGE=VARNUM INVEST GROUP,
PROJKEY=P999104)
%SAMPLE

```

Output of the example data

Test Drug A (NMP1 999) Protocol 104
Inv: Smith {023}

CONCENTRATION OF THEOPHYLLINE (ug/ml) IN SERUM

TREATMENT: GROUP 1 (THEO-24*) WITHOUT TEST DRUG A

SUBJ	0	0.25	0.50	1.0	1.5	2.0
2	3.05	3.09	2.84	2.98	3.10	3.03
3	3.67	3.80	3.63	3.63	3.57	3.67
4	1.22	1.14	1.00	1.07	1.10	1.23

Test Drug A (NMP1 999) Protocol 104
Inv: Smith {023}

CONCENTRATION OF THEOPHYLLINE (ug/ml) IN SERUM

TREATMENT: GROUP 1 (THEO-24*) WITH TEST DRUG A

SUBJ	0	0.25	0.50	1.0	1.5	2.0
2	2.89	2.95	2.87	3.02	3.38	3.71
3	4.78	4.78	4.52	4.86	5.19	5.48
4	2.04	1.60	1.58	1.60	1.78	2.10

Test Drug A (HRPI 000) Protocol 104
Inv: Smith (023)

CONCENTRATION OF THEOPHYLLINE (ug/ml) IN SERUM

TREATMENT: GROUP 2 (THEO-DUR®) WITHOUT TEST DRUG A

SUBJ	0	0.25	0.50	1.0	1.5	2.0
1	4.20	4.10	4.06	4.42	4.86	5.03
5	3.86	3.92	3.80	3.66	3.70	4.07
8	4.32	4.40	4.50	4.93	4.62	4.55

Test Drug A (HRPI 000) Protocol 104
Inv: Smith (023)

CONCENTRATION OF THEOPHYLLINE (ug/ml) IN SERUM

TREATMENT: GROUP 2 (THEO-DUR®) WITH TEST DRUG A

SUBJ	0	0.25	0.50	1.0	1.5	2.0
1	3.64	3.87	4.34	5.16	5.00	5.03
5	3.62	3.41	4.06	5.06	5.53	5.22
8	5.08	4.80	5.03	5.02	4.84	5.04

Test Drug A (HRPI 000) Protocol 104
Inv: Jones (045)

CONCENTRATION OF HRPI 000 (ng/ml) IN PLASMA

TREATMENT: GROUP 1 (THEO-24®)

SUBJ	0	0.25	0.50	1.0	1.5	2.0
2	0.0	0.1	34.8	52.2	103.1	188.1
3	20.2	25.0	30.7	215.1	310.1	488.1
4	11.5	11.0	13.0	26.0	86.8	130.1

Test Drug A (HRPI 000) Protocol 104
Inv: Jones (045)

CONCENTRATION OF HRPI 000 (ng/ml) IN PLASMA

TREATMENT: GROUP 2 (THEO-DUR®)

SUBJ	0	0.25	0.50	1.0	1.5	2.0
1	15.5	12.4	12.8	10.4	13.2	18.5
5	12.7	9.2	9.9	18.6	21.8	40.5
8	13.5	15.1	11.2	48.0	72.1	145.0

Conclusion

As is evident in the example given above, the report description information, the headings, titles, subtitles, and format specification can be handled as a data dependent system. One great advantage of such a system is that the report description information always matches the body of the report. Integrity of the reports and efficiency of the operations are two natural benefits derived from this kind of system.

Appendix

```

***** PARTSET (PMT=1,
PAGE=GROUP,
PICKVAR=1,
PROJKEY=000104);
*****
GLOBAL XPAGE;
KLET BLK=XSTR;
KLET XPAGE=XPAGE;
*****
* THIS SECTION SETS UP GLOBAL MACRO VARIABLES FOR LATER USAGE
*****
KLET DSTR=8BLK;
KLET PSTR=8BLK;
KLET SSTR=8BLK;
KLET USTR=8BLK;
KLET IX=1;
KLET VNUM=XSCAN(8PICKVAR,8IX,8BLK);
XDO N=1;LE(KLENGTH(VNUM)>0);
KLET DSTR=8DSTR.8BLK.08VNUM;
KLET PSTR=8PSTR.8BLK.08VNUM;
KLET SSTR=8SSTR.8BLK.08VNUM;
KLET USTR=8USTR.8BLK.08VNUM;
KLET IX=XEVAL(8IX+1);
KLET VNUM=XSCAN(8PICKVAR,8IX,8BLK);
XEND;
KLET ZSTR=8DSTR.8BLK.8PSTR.8BLK.8SSTR.8BLK.8USTR;
GLOBAL 8ZSTR;
*****
* THIS SECTION CREATES THE MACRO VARIABLES FROM THE DATA SET P005
*****
DATA P005;
SET OS=P005;
IF PROJKEY=8PROJKEY;
DATA _NULL_;
SET P005;
IF DECM=
THEN CALL SYMPUT('D'!!LEFT(VARNUM),LEFT(DECM));
ELSE CALL SYMPUT('D'!!LEFT(VARNUM),LEFT(0));
IF MEDIUM=
THEN CALL SYMPUT('M'!!LEFT(VARNUM),PLASMA);
ELSE CALL SYMPUT('M'!!LEFT(VARNUM),LEFT(MEDIUM));
IF SENSIT=
THEN CALL SYMPUT('S'!!LEFT(VARNUM),1);
ELSE CALL SYMPUT('S'!!LEFT(VARNUM),LEFT(SENSIT));
IF UNIT=
THEN CALL SYMPUT('U'!!LEFT(VARNUM),UGML);
ELSE CALL SYMPUT('U'!!LEFT(VARNUM),LEFT(UNIT));
RUN;
*****
* THIS SECTION SELECTS THE RELEVANT INFORMATION FROM DATA SET P005,
* GENERATES THE CODE FOR FORMAT, AND EXECUTES THE CODE TO PROVIDE
* THE FORMAT FOR A PARTICULAR PROJECT AND PROTOCOL
*****
DATA P005;
SET OS=P005;
IF PROJKEY=8PROJKEY THEN DO;
IF PHT=8PHT OR PHT=9
THEN OUTPUT;
END;
KLET CNT=0;
DATA _NULL_;
SET P005 END=EOF;
BY PHTNAM PHTVAL;
LENGTH CSTR 800;
RETRN IX 0 JX 0;
IF FIRST.PHTNAM THEN DO;
IX=1;
CSTR=VALUE'!' !!LEFT(PHTNAM);
CALL SYMPUT('STR'!!TRIM(LEFT(IX))!!',TRIM(CSTR));
JX=0;
END;
JX=1;
CSTR=TRIM(LEFT(PHTVAL))!!' '!!TRIM(TEXT)!!' ';
PUT CSTR;
IF LAST.PHTNAM THEN DO;
CSTR=TRIM(CSTR)!!';
CALL SYMPUT('CNT'!!TRIM(LEFT(IX)),TRIM(LEFT(JX)));
END;
CALL SYMPUT('STR'!!TRIM(LEFT(IX))!!TRIM(LEFT(JX)),TRIM(CSTR));
IF EOF
THEN CALL SYMPUT('CNT',TRIM(LEFT(IX)));
RUN;
REF SCNT=0 XTHEN XDO;
PROC FORMAT;
XDO IX=1 XTO 8CNT;
8ASTR8IX.0
XDO IX=1 XTO 8CNT/8IX;
8ASTR8IX8IX
XEND;
XEND;
XEND;
*****
***** PARTSET;

```

```

*****
* NAME: FINDX
* FUNCTION: TO SEARCH FOR A PATTERN IN A MACRO VARIABLE &&STRING
* LATEST UPDATE: 10/08/87
*****
%MACRO FINDX(PATTERN=INVEST,
             STRING=PAGE);
LOCAL B;
IF LENGTH(&&STRING) < LENGTH(&PATTERN)
  %THEN %LET B=0;
%ELSE %LET B=%INDEX(&&STRING,&PATTERN);
%END FINDX;

*****
* NAME: GETINFO
* FUNCTION: TO GET THE INFORMATION FOR THE HEADINGS
* LATEST UPDATE: 09/13/88
*****
%MACRO GETINFO(DATASET=LAST,
               HEADCOL=70);
%GLOBAL HEADER1 HEADER2 HEADER3 TITLES;
PROC CONTENTS DATA=&DATASET OUT=DSINFO NOPRINT;
DATA NULL;
  SET DSINFO (OBS=1 FIRSTOBS=1);
  CALL SYMPUT('DATASET',LEFT(HEWNAME));
RUN;
PROC SORT DATA=&DATASET OUT=UTIL;
  BY PROJECT PROTOCOL;
DATA UTIL;
  SET UTIL;
  BY PROJECT PROTOCOL;
  IF FIRST.PROTOCOL;
DATA UTIL;
  MERGE UTIL (IN=OK) DS.P001;
  BY PROJECT;
  IF OK;
DATA UTIL;
  MERGE UTIL (IN=OK) DS.P002;
  BY PROJECT PROTOCOL;
  IF OK;
  %IF %INDEX=0 %THEN %DO;
  DATA UTIL;
  MERGE UTIL (IN=OK) DS.P003;
  BY PROJECT PROTOCOL;
  IF OK;
  DATA NULL;
  SET UTIL END=EOF;
  LENGTH ISTR $ 120;
  RETAIN ISTR;
  ISTR=SUBSTR(ISTR,1,LENGTH(ISTR))||' '||LEFT(INVEST);
  IF EOF
  %THEN CALL SYMPUT('INVLIST',LEFT(ISTR));
RUN;
%GLOBAL INVLIST;
%END;

DATA NULL;
  SET UTIL END=EOF;
  LENGTH ISTR $ 120;
  %IF %INDEX=0 %THEN %DO;
  ISTR=ISTR||' '||LEFT(INVLIST)||' '||PUT(INVEST,Z3.);
  CALL SYMPUT('I'||LEFT(INVEST),ISTR);
  %END;
  IF EOF %THEN %DO;
  IF PROT=
  %THEN PROT=LEFT(PROTOCOL);
  SELECT (HPMT);
  WHEN (1) %DO;
  SELECT (REGTK);
  WHEN ('N')
    ISTR=TRIM(DNAME)||' Protocol '||
    LEFT(PROT);
  WHEN ('R')
    ISTR=TRIM(DNAME)||' Protocol '||
    LEFT(PROT);
  WHEN ('I')
    ISTR=TRIM(DNAME)||' Protocol '||
    LEFT(PROT);
  OTHERWISE
    PUT 'INVALID CODE FOR REGTK';
  %END;
  WHEN (2) %DO;
  SELECT (REGTK);
  WHEN ('N')
    ISTR=TRIM(DNAME)||' ('||TRIM(DNAME)||') Protocol '||
    LEFT(PROT);
  WHEN ('R')
    ISTR=TRIM(DNAME)||' ('||TRIM(DNAME)||') Protocol '||
    LEFT(PROT);
  WHEN ('I')
    ISTR=TRIM(DNAME)||' ('||TRIM(DNAME)||') Protocol '||
    LEFT(PROT);
  OTHERWISE
    PUT 'INVALID CODE FOR REGTK';
  %END;
  OTHERWISE
    PUT 'INVALID CODE FOR HPMT';
  %END;
  %SPECIAL
  CALL SYMPUT('TITLE0',ISTR);
%END;

%*%
%LET HEADER1=%STR(HTXT='&TITLE0');
%LET HEADER2=%STR(PUT @&HEADCOL HTEXT);
%IF %INDEX=0
  %THEN %LET HEADER2=%STR(ICODE=' '||LEFT(INVEST);
  HTXT=%SYMPUT(ICODE);
  PUT @&HEADCOL HTEXT);
%ELSE %LET HEADER2=%STR(PUT);
%LET HEADER3=%STR(PUT);
DATA DSINFO;
  SET DSINFO;
%END GETINFO;

```

```

*****
* NAME: PUTINFO
* FUNCTION: TO WRITE THE HEADINGS USING THE INFORMATION FROM GETINFO
* LATEST UPDATE: 09/14/88
*****
%MACRO PUTINFO;
%HEADER1
%HEADER2
%HEADER3
%END PUTINFO;

*****
* NAME: FRMT
* FUNCTION: TO GENERATE THE FORMAT
* LATEST UPDATE: 01/04/89
*****
%MACRO FRMT;
IF DEC=0 %THEN %DO;
  FRMO=TRIM(LEFT(LEN+0))||'.';
  FRM1=TRIM(LEFT(LEN+0))||'.';
  FRM2=TRIM(LEFT(LEN+2))||'.';
%ELSE %DO;
  FRMO=TRIM(LEFT(LEN))||'.'||LEFT(DEC);
  FRM1=TRIM(LEFT(LEN-DEC-1))||'.';
  FRM2=TRIM(LEFT(LEN+1))||'.'||LEFT(DEC+1);
%END;
%END FRMT;

*****
* NAME: PRINTV
* FUNCTION: TO PRINT THE VALUE IN THE CORRECT FORMAT
* LATEST UPDATE: 04/20/88
*****
%MACRO PRINTV(V=0);
SELECT (V);
  %DO K=1 %TO %N(V);
  WHEN(&K)
    %DO;
    OTHERWISE
      PUT @COL 'XXXXXXXX' @;
    %END;
%END;
%END PRINTV;

```

References

ISPF Version 2 Release 2 Dialogue Management Services, Publication Number SCJ4-4021-1, Copyright IBM Corporation 1984, 1985.

SAS Institute Inc. SAS User's Guide: Basics, Version 5 Edition, Cary, NC: SAS Institute Inc., 1985.

ISPF Dialogue Management Services is an IBM program product.

SAS is a registered trademark of SAS Institute Inc., Cary, NC, USA.