

Managing Multiple SAS/FSEDIT® Data Entry with a SAS/AF® Application For a Cohort Study

Emily A. Mixon, UAB, Birmingham, AL; Karen B. Fowler, UAB, Birmingham, AL

ABSTRACT

The University of Alabama at Birmingham, Department of Pediatrics, Division of Infectious Diseases currently follows a cohort of children diagnosed with congenital cytomegalovirus (CMV) infection. Congenital CMV infection is one of the leading causes of sensorineural hearing loss in young children. Vision, neurologic, and developmental deficits may also occur in children with congenital CMV infection. In order to determine sequelae following CMV infection, each study participant receives hearing, vision, and psychometric testing at various intervals during the followup period. Demographic information along with a maternal questionnaire and medical record abstracts are also collected for determining the possible maternal exposures for infection. SAS (ver 8) FSEDIT7 data entry screens are used for entering the information gathered into separate data sets. In order to organize and simplify the data entry process a SAS/AF7 application was developed allowing the user to select the data set needed by pressing the appropriate icon on the frame which then calls up the FSEDIT screens for the data set selected. The user is also able to print out a listing of observations added to the data set or a listing of observations with corrections made on a user specified day for monitoring data quality.

INTRODUCTION

Congenital cytomegalovirus (CMV) infection remains a major public health problem in the United States because of its frequency and its role as a cause of hearing loss, cognitive impairment, cerebral palsy and visual impairment in young children. It has been estimated that there are approximately 40,000 births with congenital CMV infection in the U.S. each year. Of these infants, 15% will have some CNS impairment due to CMV making CMV infection a leading cause of sensorineural hearing loss and the leading infectious cause of CNS damage in children in the U.S. The Department of Pediatrics, Division of Infectious Diseases at the University of Alabama at Birmingham has a large registry of children diagnosed with congenital CMV infection and follows this cohort to study outcomes following infection. Each child receives hearing, vision, and psychometric testing at various intervals during the follow-up period. A maternal questionnaire and medical record abstracts are also collected for each patient to determine possible maternal exposures for infection.

In order to make the data entry process more

efficient and cost effective, we developed a SAS/AF7 FRAME application in Version 8.0 of the SAS System on the Windows 2000 platform. The user selects the data set of choice from the main menu of the Congenital CMV Data Entry Application and customized data entry screens are invoked (Fig. 1). We use SAS/FSEDIT® for entering the data collected since complex screens with extensive SAS Component Language (SCL) were already in use for data entered on a regular basis. The application also allows the user to print out a listing of all the observations entered into each dataset providing documentation for data management purposes. This paper will describe the application and SCL used for data entry.

DESCRIPTION OF APPLICATION

The opening frame of the application presents the user with a menu of all the data set options for the Congenital CMV research project. The purpose of the Main Menu frame in the Congenital CMV Data Entry Application is to either allow the user to select the data set needed for data entry or choose to print a listing of the data entered or updated (Fig.1). The main menu frame contains Push Button Controls for selecting the data sets, a Graphic Text Control for labeling the frame and a Command Push Button for exiting the application.

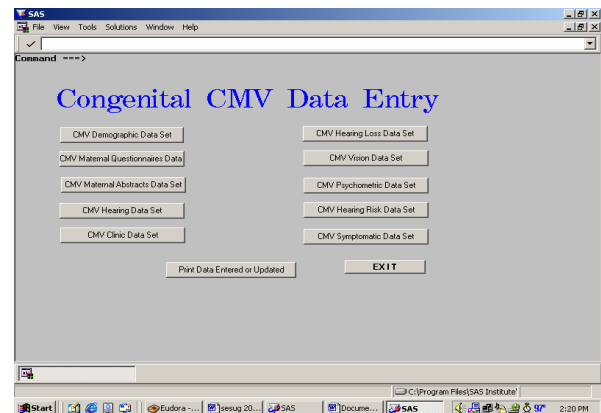


FIGURE 1

The SCL for the Main Menu frame uses the CALL FSEDIT routine to open the data set and FSEDIT screen selected by the user. If the user chooses the “Print Data Entered” button, the SCL sends the user to the Congenital CMV Data Monitoring frame using the CALL DISPLAY routine.

```

INIT:
RETURN;

BUTTON1:
CALL FSEDIT('CMV.CMV', 'CMV.CMV.CMV');
RETURN;

BUTTON2:
CALL
FSEDIT('CMV.CMVQUESTIONNAIRE', 'CMV.CMVQUESTIONNAIRE.QUE
S_ENTRY');
RETURN;

BUTTON3:
CALL FSEDIT('CMV.MRABSTRACTSCMV',
'CMV.MRABST.MRABSTRACTSCMV');
RETURN;

BUTTON4:
CALL DISPLAY('MONITOR.FRAME');
RETURN;

BUTTON5:
CALL FSEDIT('CMV.CLIN2002', 'CMV.CLINIC.CLIN2002');
RETURN;

BUTTON6:
CALL FSEDIT('CMV.NEWHEAR', 'CMV.HEARING.NEWHEAR');
RETURN;

BUTTON7:
CALL FSEDIT('CMV.SNHL', 'CMV.SNHL.SNHL');
RETURN;

```

The Data Monitoring frame is a menu frame similar to the main menu of the CMV Data Entry frame, with Push Button Controls to select the data set for printing a listing of data entered (Fig.2).

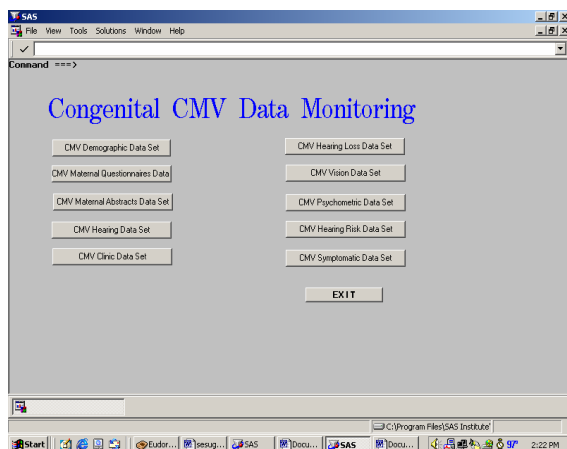


FIGURE 2

When the user selects a data set, such as the CMV Maternal Questionnaires Data, the CMV Maternal Questionnaire Data Set frame is displayed using the CALL DISPLAY routine (Fig. 3). The user enters the date for the print out in Input Fields and once the “Print” Desktop Icon Control is pressed the SCL for the frame is initiated.

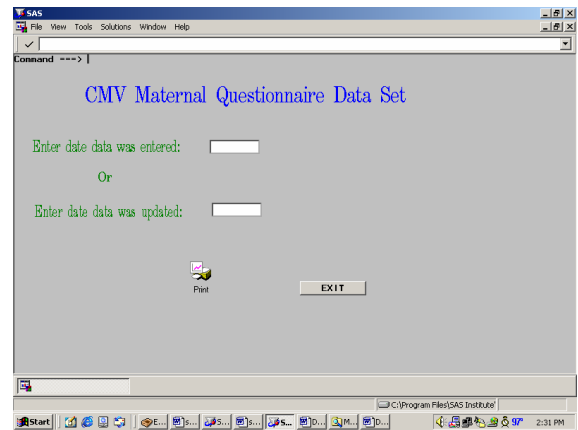


FIGURE 3

Within a SUBMIT CONTINUE routine, the data is subset by the date entered by the user and the ODS PRINTER statement is used to open and manage the printer destination. The PROC PRINT statement sets the specifications for printing the output and a title is generated using the data entry date entered by the user. The variables selected are listed and then the ODS PRINTER CLOSE statement is issued followed by the ENDSUBMIT command. Similar SCL is used for printing a listing of data updated for a selected date. When the “Print” Desktop Icon is pressed, the data is subset by the selected update date and then a listing of all the variables for each observation updated in the database on that date is printed.

```

RC=RC;
INIT:
RETURN;

PRINT:
IF EDATE NE _BLANK_ THEN DO;
PUT 'EDATE';
PUT ENTRYDAT= EDATE=;

SUBMIT CONTINUE;
OPTIONS CENTER PAGENO=1 PS=52 NODATE;
LIBNAME ENTRY 'C:\DATA\CMV';
DATA TEMP; SET ENTRY.CMVQUESTIONNAIRE;

TITLE ' ';
TITLE2 ' ';
DATE_1=&EDATE;
CALL SYMPUT('DT1',PUT(DATE_1,MMDDYY8.));
RUN;

DATA A; SET TEMP;
IF ENTRYDAT='&EDATE';

ODS PRINTER;

PROC PRINT D NOOBS WIDTH=MINIMUM; TITLE "MATERNAL
QUESTIONNAIRE DATA SET FOR DATA ENTRY DATE = &DT1";
TITLE2 'MATERNAL QUESTIONNAIRE';
VAR STUDY_ID DOB DOI MDOB MRACE MMARITAL GRADE HS GED
COLLEGE COLL_YRS;
RUN;

```

FSEDIT SAS COMPONENT LANGUAGE

Extensive SCL is used behind the FSEDIT screens for providing data quality. External SAS data sets are used for verifying information entered by the user, such as ID numbers and names and also for linking data residing in the external data sets. For example, given a code for a type of medication, the description of the medication and the American Hospital Formulary category is pulled into the data set and onto the screen from the secondary SAS data set. This allows for validation of the data during the data entry process. The SCL for the Maternal Questionnaire FSEDIT screens is described below. The SCL for the other data entry screens is similar and will not be further discussed in this paper.

In the FSEINIT section of the code, librefs are assigned and several data sets are opened for validation of the following variables: occupation, employment, medication type, American Hospital Formulary category and reasons for medication. Also, the data set for verifying the ID number and linking the name of the patient is opened in the FSEINIT section of the SCL.

```
fseinit:

if (libname('CMV','c:\data\CMV')) then _msg_=sysmsg();
occ=open('CMV.occupcode', 'u');
emp=open('CMV.emplcode', 'u');
otc=open('CMV.otccodes', 'u');
pre=open('CMV.prescodes', 'u');
anti=open('CMV.anticodes', 'u');
pr1=open('CMV.presreason', 'u');
antir=open('CMV.antireason', 'u');

occnum=varnum(occ, 'jobcode');
occdes=varnum(occ, 'jobdesc');
empnum=varnum(emp, 'emplcode');
empdes=varnum(emp, 'empldesc');
otcnum=varnum(otc, 'otccode');
otcdes=varnum(otc, 'otcdesc');
prenum=varnum(pre, 'prescode');
predes=varnum(pre, 'presdesc');
antinum=varnum(anti, 'anticode');
antides=varnum(anti, 'antidesc');
prernum=varnum(pr1, 'presreas');
prerdes=varnum(pr1, 'presreasdesc');
antirnum=varnum(antir, 'antireas');
antirdes=varnum(antir, 'antireasdesc');
imid=open('CMV.cmvids');
call set(imid);
return;
```

In the MAIN section of the SCL the Study ID entered by the user is searched for in the ID data set using the LOCATEC statement.

```
MAIN:
rc=locatec(imid,varnum(imid,'study_id'),study_id,'a');
if STUDY_id gt 0 then do;
if (rc=0) then do;
_msg_='There is no Study ID Number equal to '||
study_id ||' .';
alarm;
erroron study_id; end;
```

If the ID number is found then the name linked with that ID is displayed on the screen allowing the user to verify that the data being entered is for the correct individual. The LOCATEC, or LOCATEN statement if the variable being searched is a numeric, is used for all the data sets in verifying occupation, employment, medication type, American Hospital Formulary category, and reasons for medications. The date of data entry is assigned to equal the computer's system date. The CONTROL LABEL statement allows variables to be checked for modifications before the rest of the MAIN section executes. If a variable on the screen is modified, then the update date also becomes the computer's system date. Otherwise, the update date does not change.

```
if entrydat eq . then do;
entrydat=today();
end;
upd8dat1=today();
control label;
study_id;
if upd8dat1=today() then upd8dat1=upd8dat;
return;
ABNORMAL: ABNTREAT: ABN_DATE: ABN_PAP: AGE1: AGE2:
AGE3: AGE4: AGE5: AGE6:
ALC: ALC_BEG: ALC_END: ALC_NUM: ALC_PREG: ALC_TIME:
ALC_USE: ANTI1: ANTI2:
upd8dat=today();
```

The remainder of the MAIN section validates whether the values entered by the user are acceptable and reasonable for each variable based on predetermined error checking to limit the amount of erroneous data. If a code cannot be found or is not an option then the user is given a message that the code is unknown or given a list of acceptable choices.

```
if doi ne . then do;
if doi lt dob then do;
erroron doi;
alarm;
_msg_='Interview date should be > DOB';
end;
end;

if mrace not in (1,2,3,4,5,9 .) then do;
erroron mRACE;
alarm;
_msg_='Race is either 1-White, 2-Black, 3-Hispanic, 4-Asian 5-Other, 9-Unk';
end;
if mmarital not in (1, 2, 3, 4, 5, 6, 7, .) then do;
erroron mmarital;
alarm;
_msg_='Marital Status is either 1-Single, 2-Married, etc.';
end;
```

Also, code that validates the logic or consistency of variables such as verifying the date entered for last menstrual period is less than the due date or the number of pregnancies is consistent to the number of pregnancy outcomes is included in the MAIN section of the SCL.

```
if pregnum=0 and (preg1 ne . or preg2 ne . or preg3 ne
. or preg4 ne . or preg5 ne . or preg6 ne .) then do;
  erroron pregnum;
  _msg_='Make pregnancy numbers match';
end;
```

The FSETERM section of the SCL closes the data sets opened in the FSEINIT section and also de-assigns any librefs.

```
fseterm:
  call close (imid);
  call close (occ);
  call close (emp);
  call close (otc);
  call close (pre);
  call close (pr1);
  call close (anti);
  call close (antir);

if (libname('CMV')) then _msg_=sysmsg();
```

CONCLUSION

The Congenital CMV Data Entry provides an efficient method for maintaining the location of data sets and their related FSEDIT screens for routine data entry. This type of application is especially useful when multiple data sets are routinely updated and access similar secondary or external data sets. The application along with extensive SCL behind the FSEDIT screens makes the data entry process cost effective and improves the quality of the data. Also, the application provides documentation of data entered or changed for data monitoring. The simplicity of the application makes it potentially usable in the data management of other research studies.

REFERENCES

Birdeson, Patti M. (1996), "Creating a Contacts Application Using New Classes in SAS/AF Software". Observations, First Quarter 1996.

SAS Institute Inc. (1997), Building SCL Applications using Frame Entries Course Notes, Cary, NC: SAS Institute Inc.

ACKNOWLEDGMENTS

This work was supported in part by the National Institutes of Health, National Institute of Allergy and Infectious Diseases Grants U01 AI41025 and P01 AI43681. The Authors would also like to thank Ann Carpenter Rockett at SAS Technical Support for her helpful advice and suggestions in the development of the SCL in this application.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Emily Mixon

UAB Department of Pediatrics

CHT 752, 1600 7th Ave. South

Birmingham, AL 35233

Work Phone: 205-939-6687

Fax: 205-975-3221

Email: emixon@uab.edu