A SAS/FSP SOFTWARE AND CLIST SYSTEM FOR EASY MANAGEMENT OF A MACROINVERTEBRATE REFERENCE COLLECTION

Todd C. Folsom, Duke Power Company

Introduction

The base SAS® software, SAS/Full Screen Product, and IBM's CLIST command language can be linked to form a menu-driven system that allows non-programmers to handle all tasks for management of SAS datasets. These tasks include data entry, data checking, appending of new data and report generation. This paper shows how such a system was designed to manage data from a collection of macroinvertebrates.

Biologists doing environmental monitoring work are often required by regulatory agencies to support the taxonomic identifications the biologists make. When such identifications have been verified by taxonomic authorities, the specimens are very useful for comparison with new, unknown specimens. A comprehensive, up-to-date list of specimens is essential for biologists to have if they are to make effective use of such a collection. Therefore we computerized the specimen information for a collection of freshwater macroinvertebrates that supports our monitoring studies.

This paper is arranged to first show how the data base was set up, protected, and self-documented. Next I show how to add and check new data using a CLIST program and SAS/Full Screen Product (SAS/FSP®). Then appending the new data to the permanent dataset and producing quality-assurance documentation is covered. The report-writing program is briefly discussed.

Design of the Dataset

A major consideration in designing the dataset was to keep the storage requirements as low as possible. One way to save storage space is to reduce the 8 byte length of numeric variables to the minimum needed. I set the length of the variables MALES and FEMALES (Fig. 1) to 2 bytes, which can accommodate integer values large enough for our needs. Seven-digit taxa codes paired with Latin species names were already available in a disk file as part of our processing scheme for study data. Therefore, taxa codes were stored, rather than the much longer taxa names, because the names and codes can be read from the taxa file and merged with the codes on the collection dataset to provide the Latin names.

The initially available data were read from cards, saved permanently in a SAS library under the member name REFCOLL and labeled Macroinvertebrate Reference Collection (Fig. 1), taking advantage of some of the self-documenting features of SAS (Merlin 1984). The data are write-protected with the PROTECT=XXXX option, where XXXX is a password.

PROC CONTENTS displays the directory information for the dataset REFCOLL (Fig. 1). This shows the variables and their formats, when the dataset was created, its descriptive label, and the source SAS statements. Note that the password is not listed. The password, as well as other attributes of the dataset, can be altered by PROC DATASETS.

The variables STAGE, PREP, and VERIF refer to the life history stage of the specimen, the kind of specimen preparation used and whether the identification has been verified by an expert. These character variables were given a length of one on the dataset. It would enhance the clarity of data reports to have these variables formatted. PROC FORMAT was used to turn "A" in Adult, "L" into Larva and so on. The formats were stored in a permanent SAS format library. When this file is allocated to the DD name "SASLIB", and a FORMAT statement with one of the stored format names is used in a PROC step, the system will automatically look for and use the specified format in the SASLIB file.

Working With the Dataset

New data are put into a temporary data set under the user's TSO id, then run through a check program to spot errors. If errors are found, the data are edited and rechecked. When the data are correct, they are appended to the permanent on-line file. If any changes need to be made to the permanent file, the SAS/FSP editor is used to make them directly on the file.

PROC FSEDIT permits customized screens to be set up and stored for future use. I arranged the variable names on the screen in the order they are on the coding sheet for ease of data entry (Lafler 1984). Dashes follow each variable name and indicate the allowed length of each variable. Under the dashes after DATE I put ddMMMyyyy to indicate the format each date value must be in, e.g. 23FEB1984. I made TAXA, STATE, and PREP required variables, so a new observation will not be accepted by SAS/FSP unless the observation contains a value for each. Those variable names are highlighted in cyan color for good visibility (Lafler 1984). STATE

abbreviations will be automatically capitalized, but COUNTY and LOCALITY values can be lower case. All of this is taken care of by the customized FSEDIT screen.

Adding and Checking New Data

The process of adding, validating, editing, and appending new data is controlled by a CLIST program. CLIST is short for Command List, and is a language that can contain TSO commands and SAS code, as well as CLIST statements. This CLIST program writes a menu and other instructions on the screen, receives the user's responses and controls the action through executing TSO commands and invoking the SAS system.

Usually the first step is to type in new data. The CLIST and SAS set up a new, empty file under the user's TSO id to receive the data. After the new data are typed in using PROC FSEDIT, they are run through a validation program to print the data. If corrections must be made to the data, the CLIST will let the user cycle through the FSEDIT step and the validation step until they are satisfied that the data are correct. Then the CLIST uses PROC APPEND to add the new data to the permanent dataset, run PROC CONTENTS, and list the new data on paper. This final step was designed to produce most of our required quality assurance documentation automatically.

The Menu-- The CLIST writes some introductory notes on the screen and then lists a menu like this:

```
A. Add and check new data (do
initially).
B. Edit and recheck recently added
data (can do repeatedly).
C. Append new data to the permanent
data set and delete the 'new' data
set (do last).
D. List the permanent data set (N>
3000 lines) via batch (do anytime).
E. Quit.
Type an A through E to make your
choice:
```

Quitting will return to READY mode in TSO. Options A to C will be discussed next.

Adding Data-- The CLIST performs allocations of the necessary files and puts the user into interactive SAS (Fig. 2). The empty dataset is prepared for use through the DATA step where the variables are identified in the LENGTH statement and given informats where desirable. Then SAS/FSP starts and provides the customized data entry screen. When all the observations have been entered the check program will run automatically through the %INCLUDE statement. It will read the taxa name file and merge the two files by taxa code. Then the collection data will be printed by species using the predefined formats.

The data are examined for missing species names which indicate invalid taxa codes. Unformatted values of STAGE, PREP, and VERIF indicate wrong values. The STATE, COUNTY, LOCALITY, DPCSTA, DATE, MALES, and FEMALES data are matched with values on the coding forms. If there any errors, option B is chosen from the CLIST menu, otherwise the user can go straight to option C.

Edit and Recheck Data-- Option B is more flexible in what it allows, but requires more work to control the action . The user will be put into interactive SAS after being told on the screen what to type to accomplish their mission. This option makes use of some nice features of SAS that will let the tasks run faster and save a little time.

These statements are written on the screen before the CLIST executes SAS:

```
Type %INCLUDE CHECK; to run the
check program.
Type %INCLUDE FSED; to go to SAS
FSEDIT.
Type %INCLUDE RECHECK; to re-run
the check program.
Type ENDSAS; to end. Make a note of
these!
When the screen says 'READY' after
SAS ends, type CONTINUE to go on.
```

Usually the first step will be to go into FSEDIT and make the corrections to the data detected with the run of the check program after initial entry of the data. After doing this and ending FSEDIT, the user should check that the corrections were correct, so %INCLUDE CHECK; does that. The %INCLUDE tells SAS to find and run the program allocated to the Data Definition (DD) name CHECK. The three DD names, CHECK, FSED, and RECHECK are stored programs in a permanent disk file.

If there are still errors to correct, go to FSEDIT. To recheck the latest corrections a RECHECK program is used that takes advantage of the fact that the SAS system remembers all data sets it previously created as long as the SAS session has not ended. Thus the data set containing over 5000 taxa names from the taxa file does not have to be re-created as the CHECK program would do. The RECHECK program simply calls in the newly edited data and proceeds with the merging of the existing work dataset of names and taxa codes with the collection dataset and then printing the resulting data.

When the data are correct, the SAS session is ended. The computer will print READY after you end SAS, but the user will want to get back into the

CLIST program. The CLIST statement TERMIN CONTINUE will cause the CLIST will resume control when the user types CONTINUE.

Appending Data-- Option C is the final step, and is where the new data are appended to the end of the permanent dataset (Fig 3). However a password must be supplied to the PROTECT= option attached to the permanent dataset. The user is prompted for the password by the CLIST, and it is read in as &PASS. When the SAS system is invoked, the &SYSPARM function reads &PASS and inserts the password into the PROTECT= option. The password is not listed in the CLIST source program, so it is completely secret.

PROC APPEND adds the new data onto the old dataset and then prints notes about how many observations were added. PROC CONTENTS is run to display the latest characteristics of the REFCOLL dataset. A printout of the data appended follows. The date was printed (because the DATE option was specified when SAS was invoked), so all our Data Clerk has to do is initial the printout and add it to our quality assurance log for data processing, to document his actions.

The CLIST asks if the append job ran okay. The user examines the SAS log on the printout for any error messages. If they appear, the user types NO at the prompt. If the job ran ok, YES is typed and the CLIST will delete the temporary data set. This prevents an error from occurring if the user tried to choose Option A (add new data) before the dataset had expired from their TSO id.

Writing the Data Report

Option D submits a batch job that writes a report listing the data in a very readable format. The first step in the program is to read the family and species names from the taxa file and associate the proper family name with each species name. Then after bringing in the REFCOLL data, each taxa code is associated with its species and family names. The report is written using FILE PRINT for precise formatting. Each time a new family name occurs in the file, the name is printed at the left margin (Fig. 4). Likewise, each new species name is printed along with its taxa code. The data are listed for each species, and the stored formats are used for the appropriate variables.

An additional listing was requested by our curator that would show which species were contained in the collection by printing just species names under their family names. This provides a quick reference for determining whether a species is in the collection without having to page through the much larger master printout. The program takes advantage of "FIRST.byvariable" processing to select only the first ocurrence of a new species name for printing.

Custom reports can also be written that list certain taxonomic groups, like Odonata or Diptera, when subsetting IF statements are used to specify the range of taxa codes to read from the REFCOLL file. Reports that show the species collected from certain localities can be prepared by using subsetting IF statements to only read in the state or county records of interest from the REFCOLL file. Distribution maps of a species by county or state can be prepared using the SAS/GRAPH® map datasets.

References

1. Lafler, K.P. 1984. Human factors engineering in SAS/FSP applications. SAS Institute, Ed. SAS Users Group International Conference Proceedings. Cary, NC. p. 305.

2. Merlin, R. 1984. Design concepts for SAS applications. SAS Institute, Ed. SAS Users Group International Conference Proceedings. Cary, NC. p. 283.

Todd C. Folsom
Duke Power Company
Production Environmental Services
Rt. 4, Box 531
Huntersville, NC 28078

Figure 1. Condensed Output of PROC CONTENTS.

```
Contents of SAS data set DD.REFCOLL
Observations=2308  Label=
Macroinvertebrate Reference Collection.

Alphabetic list of variables
 #  Variable   Type  Length  Position
       Label

 3  County     Char    15       13

11  Date       Num      8       73

 5  Dpcsta     Char     5       61
       Duke station no.

 8  Females    Num      2       69

 4  Locality   Char    33       28

 7  Males      Num      2       67

 9  Prep       Char     1       71
       Specimen preparation

 6  Stage      Char     1       66
       Life history stage

 2  State      Char     2       11

 1  Taxa       Char     7        4
       Taxa code

10  Verif      Char     1       72
       Identification verified
```

Figure 2. Portion of the CLIST program for menu choice A. On the left margin are letters indicating the nature of the statement: C means CLIST, T means TSO, and S means SAS.

```
C  OPT1: IF &TYPE=A THEN +
C        DO
T  FREE F(IN,BUGS,SASLIB,CHECK)
T  ALLOC DA('DK80.ENV.SASFMT') F(SASLIB)SHR
T  ALLOC DA('DK80.BENTHICS.ISAM.MASTER') F(ISAMIN)SHR
T  ALLOC DA('DK80.ENV.ZOOLOGY(BENCHECK)') F(CHECK)SHR
T  ALLOC DA('DK80.ENV.ZOOSAS') F(IN)SHR
C  WRITE WHEN *** ARE SEEN AFTER SAS IS INVOKED, PRESS ENTER
C  TO SEE FSP
S   SAS OPTIONS('CLIST NOCAPS') SHARE
C  DATA
S   TSO ALLOC DA(BUGFILE) F(BUGS)NEW;
S     DATA BUGS.NEWOBS;
S      LENGTH TAXA $ 7 STATE $2 COUNTY $15 LOCALITY $ 33
S     DPCSTA $5 STAGE $1 MALES 2 FEMALES 2 PREP $1
S     VERIF $1 DATE 8;
S     INFORMAT DATE DATE9.; FORMAT DATE DATE9.;
S      STOP;
S  PROC FSEDIT DATA=BUGS.NEWOBS SCREEN=IN.COLLSCR;RUN;
S  %INCLUDE CHECK;
S   ENDSAS;
C  ENDDATA
```

Figure 3. Portion of the CLIST program for menu option C.

```
C  OPT3: IF &TYPE=C THEN +
C    DO
C  WRITENR ENTER PASSWORD:
C  READ &PASS
T  ALLOC DA('DK80.ENV.ZOOSAS') F(IN)OLD
T  ALLOC DA(BUGFILE) F(BUGS)SHR
S  SAS OPTIONS('CLIST NOCAPS DATE SYSPARM=''''&PASS''''')
C SHARE
C DATA
S   PROC APPEND BASE=IN.REFCOLL (PROTECT=&&SYSPARM)
S        DATA=BUGS.NEWOBS;  RUN;
S   PROC CONTENTS DATA=IN.REFCOLL HISTORY; RUN;
S   PROC PRINT DATA=BUGS.NEWOBS;
S   VAR TAXA STATE COUNTY LOCALITY DATE STAGE MALES
     FEMALES PREP VERIF;
S   FORMAT DATE DATE9.;
S   TITLE 'DATA ADDED TO REFERENCE COLLECTION'; RUN;
S  ENDSAS;
C   ENDDATA
C  AGAIN:WRITE DID THE APPEND JOB RUN OK?
C WRITE IF YOU TYPE YES, I WILL DELETE THE 'NEW' DATA SET.
C  WRITE IF YOU TYPE NO, I WILL GIVE YOU THE MAIN MENU.
C  WRITENR WHAT IS THE ANSWER?:
C  READ &ANS
C     IF (&ANS^=YES AND &ANS^=NO) THEN GOTO AGAIN
C    IF &ANS=YES THEN DO
T    FREE F(IN,BUGS)
C /*BUGFILE IS THE TEMPORARY DATA FILE */
T    DELETE BUGFILE
C  WRITE REMEMBER TO UPDATE THE QC LOG! BYE.
C   END
```

Figure 4. Portion of the data report as produced on a laser printer.

MACROINVERTEBRATE REFERENCE COLLECTION 15:18 WEDNESDAY, JANUARY 22, 1986

STATE	COUNTY	LOCALITY	DPCSTA	DATE	STAGE	MALES	FEMALES	PREP	VERIF
TETRAGONEURIA SPINIGERA		6897035							
MN	ST. LOUIS	SOUDAN		04JUL1984	ADULT	1	.	ENVEL.	
NY	Hamilton	Long L.		01JUL1985	ADULT	.	1	ENVEL.	YES
ON		GIRARD L. N OF NESTOR FALLS		10JUL1984	ADULT	1	2	ENVEL.	
WI	PORTAGE	SEVERSON L.		15MAY1972	LARVA	2	.	VIAL	YES
WI	SAWYER	ROADSIDE DITCH, HWY 27		19JUN1975	ADULT	1	.	VIAL	YES
GOMPHIDAE									
DROMOGOMPHUS ARMATUS		6915005							
NC	Iredell	Lookout Shoals L.		13JUN1985	ADULT	1	.	ENVEL.	
DROMOGOMPHUS SPINOSUS		6915010							
ME	York	Acton, Mousam L.		07JUL1985	EXUVIA	.	.	VIAL	
NC	BURKE	L. JAMES, LINVILLE ARM		05AUG1983	EXUVIA	.	.	VIAL	YES
NC	Iredell	Lookout Shoals L.		13MAY1985	ADULT	1	.	ENVEL.	YES
NC	McDowell	L. James at Catawba R.		12AUG1985	ADULT	2	.	ENVEL.	
SC	ANDERSON	SALUDA R. ABOVE LEE DAM		14APR1976	LARVA	.	.	VIAL	
SC	ANDERSON	SAL. R. LEE STA. ABOVE LEE DAM		14APR1976	LARVA	.	.	VIAL	
SC	OCONEE	L. KEOWEE	504.0	03NOV1975	LARVA	.	.	VIAL	
SC	YORK	L. WYLIE	225.0	23AUG1982	ADULT	2	.	ENVEL.	YES
DROMOGOMPHUS SPP		6915000							
NC	DAVIDSON	YADKIN R. BUCK STA. DISCHARGE	430.2	18AUG1977	LARVA	1	.	VIAL	
NC	DAVIDSON	YADKIN R. BUCK STA. DISCHARGE	430.1	26AUG1977	LARVA	1	.	VIAL	
NC	DAVIDSON	BUCK STEAM STATION DISCHARGE	430.2	18AUG1977	LARVA	.	.	VIAL	
NC	DAVIDSON	BUCK STEAM STATION DISCHARGE	430.1	26AUG1977	LARVA	.	.	VIAL	
GOMPHUS CAVILLARIS		6921035							
NC	FORSYTH	BELEWS L.		11MAY1983	EXUVIA	3	.	VIAL	
NC	STOKES	BELEWS L.		29MAR1983	LARVA	.	.	VIAL	