EPA/600/A-95/147

MANAGING LARGE DATABASES WITH CUSTOMIZED SAS WINDOWS

Ronald W. Matheny, U. S. Environmental Protection Agency, Research Triangle Park, NC

ABSTRACT

This paper discusses the principles of database management through customized windows using SAS/AF, particularly PROC BUILD, to invoke interactive and batch processing of data entry, editing, updating, automatic report generation, and custom report generation functions, including graphics. SAS/ACCESS is used to efficiently store the data in ORACLE using about a third of the memory that SAS databases require. Customized "views" are developed using PROC SQL with PROC/ACCESS and BASE/SAS to efficiently manage extraction of the data from the ORACLE database into SAS for a variety of purposes. Security from both the ORACLE and SAS sides is discussed as it relates to a network of personal computers and Unix workstations and Unix servers including storage of data on different Unix machines to allow more effective controlled access to the data.

This paper also gives a detailed skeleton of the manual and automated data entry and quality assurance procedures that may be needed for proper database management before customized windows are used.

This paper has been reviewed in accordance with the United States Environmental Protection Agency's peer and administrative review policies and approved for presentation and publication. Mention of trade names or commercial products does not constitute endorsement or recommendation for use.

INTRODUCTION

Large datasets, such as used in environmental monitoring studies, are characterized by data elements related to a single sample. For example, to obtain a daily average of a compound, let us say sulfur, it is necessary to state the location of the sample, the time and date, the type of sampling device used, duplicate samples (if taken), filters and materials used in collection, analysis method and protocols, calibration of instruments, duplicate analytic processing, values derived, computerization codes, time of analysis, method of averaging, handling of non-detection levels, missing values and a host of other elements. All of these elements are not needed for every sample so it is often inefficient to use a fixed-format for the data. A relational data system is much more appropriate. ORACLE is a relational data base management system (DBMS) which links data elements together through the use of tables utilizing key variables common to each table.

Data analysis, on the other hand, is much more efficient in minimizing computer resources if it is in a fixed format usually using double precision arithmetic. Resources are saved if the data already exist in this form or the steps necessary to put the data in this form are minimized. In addition, statistical routines which use a common format are much more computer efficient as the user moves from one routine to another. This is where SAS excels.

A further difficulty, which is human related, is that of translating data in a DBMS into the form necessary for statistical analysis regardless of the statistical package. In general this requires tedious writing of extensive computer code and an extensive knowledge of both the DBMS and statistical languages.

The purpose of this short manuscript is to examine the efficiency in storage and retrieval of data, the ease of processing statistical data, and the movement of data from one system to another using only the SAS language. It gives the skeleton of a complete analysis system, manual and automated, which is very user-friendly and does not require any knowledge of ORACLE or SAS.

This paper assumes that the reader has read and understood the material in "Getting Started with the Frame Entry: Developing Object-Oriented Applications." This document introduces the Frame Entry, demonstrates how a Frame Entry is created, uses traditional and modified programs with a FRAME Entry, using subclasses and methods, and communicating with objects. It is based on a graphical user interface (GUI) approach utilizing a mouse. GUI uses window elements such as bit-mapped graphics, icons, pull-down and pop-up menus, command buttons, scroll bars and sliders. Applications are usually quicker and simpler to use than command based interfaces since all the user has to do is point-and-click on the GUI element rather than enter commands. GUI interfaces are usually more intuitive and, if written correctly, do not require outside documentation thus lessening the need for training and documentation. Another major advantage is that GUI often prevent erroneous input because they present selection options through list boxes, radio boxes, and check boxes instead of reading free format commands and fields.

Creating FRAME applications can make use of object-oriented-programming (OOP) for writing both software and applications. OOP refers to the methodology of developing software where procedures and flow is de-emphasized and the emphasis is placed on the data (the objects). In contrast to traditional programming, OOP does not concentrate on the steps the program will perform first and which steps are next. OOP concentrates on the data in the programs and the operations performed on the data. Rapid prototyping, and indeed system development, is speeded up and reduces system development time significantly.

In theory, application design therefore, consists of analyzing data and grouping them into similar categories upon which similar actions are performed. These data categories are called classes and the actions are called methods. An object is a data element that is derived from a class. OOP is probably the best tool for managing large databases where editing, browsing, subsetting, updating, and appending data is necessary. It is also excellent for managing text and graphic output and controlling system security. Specific areas covered in this paper include standards, lists, use of macros, arrays, and using SCL with SAS/AF and SAS/FSP software.

The software used is SAS/AF, SAS/FSP and SAS Screen Control Language (SCL). Other SAS products are also used such as SAS/ACCESS and SAS/GRAPH.All code was written and tested using the UNIX "C" Shell and SAS 6.10 and ORACLE7. The code was written for the UNIX platform although the same procedures may be used on any platform. The testing of the code for this paper was completed on a Data General 532 UNIX workstation utilizing SAS and ORACLE7 on a DEC Alpha 4000 running under OSF/1. The only differences between what is being presented and other platforms are in the methods used in file references for a particular operating system

This Data Management Plan discusses the basic approach to be used for both phases in the United States Environmental Protection Agency's Lower Rio Grande Valley Environmental Scoping Study (LRGVESS). It addresses, in addition to data management, quality assurance of the data after it has been examined for quality control by the analytical laboratories or field collection teams. The data access methodology is presented, and quality assurance codes are part of the database.

The data is stored in a similar manner to which it is collected. As such, the database has two main components, (1) the Main Monitoring Site and (2) the Residences. Each of these major components are further divided into smaller components. These components are called "VIEWS."

The Main Monitoring Sites components consist of at least six VIEWS: (1) Mass, (2) Acidic Gases, (3) PAHs and Pesticides, (4) Trace Metals and Scanning Electron Microscopy (SEM), (5) VOCs and GC/MS and (6) Meteorology. Samplers include VAPS, Dichotomous, PS-1, Summa canisters, MES, and Low Volume PUF. Other VIEWS are created as needed.

For the Residences components there are also six elements which are further subdivided into VIEWS as follows:

(1) Air (Indoor and Outdoor, each further subdivide into six VIEWS: (a) VOCs, (b) Pesticides, (c) PAH, (d) Trace Element, (e) Mass, and (f) Morphology).

(2) Water (divided into five categories: (a) VOCs, (b) Pesticides, (c) Metals, (d) PAHs, and (e) Microbiological).

(3) Diet (divided into three categories; (a) PAHs, (b) Metals, and (c) Pesticides).

(4) House Dust (divided into three categories; (a) PAHs, (b) Metals, and (c) Pesticides).

(5) Soil (divided into three categories; (a) PAHs, (b) Metals, and (c) Pesticides); and Human Biological (Blood, Breath, and Urine) divided in the following categories; Blood (VOCs, Metals, Pesticides), Breath (VOCs), and Urine (PAHs, Metals, and Pesticides).

The components, elements, and categories are designed so that the responsible Agency and/or contractor is able to quality assure its collected data before it is entered into the database. It is also designed to pinpoint who is (was) responsible for the original quality assurance of the data and ensure that correct procedures are implemented for quality assurance. For each data field a quality assurance tag indicates how reliable the data is. A subcomponent of the Data management System is tabular and graphical representations of the data, in particular to further ensure the quality of the data. Plots and graphs are automatically produced which will show both predicted and actual data points using regression techniques and other techniques as appropriate.

The system also stores the results of the questionnaires administered in conjunction with the residential sampling to gain information regarding housing, socioeconomic status, housing characteristics, time/ activity patterns, and diet.

Development of the Data System

It is the general policy of EPA that standard, off-the-shelf software be used whenever possible in developing systems. To comply with this requirement, the data for both phases is stored in ORACLE, the EPA standard relational database package. The ORACLE database is linked to several other standard packages including SAS. The user is able to extract from the ORACLE database appropriate data in the format needed for analysis through the off-the shelf software package, SAS, which is connected to the EPA Network . The final system will integrate through DEC Pathworks and TCP/IP, the PC DOS versions, PC windows versions, VAX versions and UNIX versions and are transparent to the user.

The Data Management System was developed in two distinct stages. Stage One was the development of a pilot system which will include a SAS/ORACLE interface. Stage Two, a full ORACLE/SAS System was ready for data entry via ASCII data files or SAS databases in December, 1994. Only data meeting predetermined quality assurance criteria is accepted into the system and the contributors of the data are responsible for any updates of the data. In both the pilot and full systems, input data is carefully screened to ensure that the data stored in the system is exactly the same as the data on the electronic media input into the system. The operating system is transparent to the user. This allows the user to see data and programs as if he/she were working on a PC or VAX and use the same command structure.   The system has a point-and-click menu driven system. If a user wishes to access and analyze data on the IBM mainframe, he/she may use the File Transfer Protocol (FTP) to transfer the data to or from the UNIX systems. This approach allows any user with access to Internet or the EPA Network or even access via a modem, anywhere in the world to use the system with proper authorization.

The system has a database administrator who assigns accounts and passwords, backups databases, loads databases, ensures that the data entering the system has been quality controlled, assists users in obtaining data and extracts data for users when necessary, and generally manages the database.

System programmers develop the interfaces based on the pilot experience and user requests. They program ORACLE VIEWS and TABLES for data extractions. Analysis, statistics, and graphics capabilities use SAS. The system programmers maintain the documentation of the system, and perform general maintenance tasks.

STAGE ONE OBJECTIVES

1. Test mechanisms which load the Air Quality data into SAS from ASCII and then into ORACLE Tables.
2. Develop quality assurance and quality control subroutines which check the data for extreme values and check to ensure that data reaches detection levels and flags exceedence levels. Quality Assurance codes are placed with each data record value. This involves scatter plots of data and tabular displays.
3. Develop a pilot graphical user interface (point-and-click screens) to view the data.
4. Develop standard report mechanisms which allow users to specify which fields they desire in reports and graphics and implement these mechanisms.
5. Develop output routines to view user selected ORACLE data through UNIX SAS and transfer user selected elements to LOTUS.
6. Develop a demonstration video, computer sideshow, or briefing to demonstrate the pilot system.

STAGE II OBJECTIVES: QUALITY ASSURANCE AND QUALITY CONTROL

DATA BASE CONVENTIONS

Units of Measure

The LRGVESS Data Base is designed to be SI-metric compatible (ASTM 1976), and all data are converted to these units when possible. Land area measurements are expressed in hectares (HA), weights in kilograms (KG), grams (g), micrograms (µg), air volumes as cubic meters (M3), and liquid volumes in liters (L). Centimeters (CM) and degrees Celsius (C) are used in the climatic sector. Chapter III provides a list of the units of measure and their CAS numbers for elements and compounds as used in the LRGVESS Data Base.

Variable Names and Labels

SAS variable names are assigned to be as meaningful and unique as possible within the LRGVESS Data Base. Names may end with a single letter indicating the variable type. For ease of administration, these names are identical to those used in the ORACLE tables.

Data Set Characteristics

Data sets are assigned a permanent ORACLE Descriptor VIEW name and are unique to the type of analyses being performed. Dictionary data VIEWS are sorted by the code variable. Identifier variables and numeric code variables are stored as character variables. Character variables are not used in numerical analysis and are usually defined with leading zeros (e.g., "001"). To conserve space on the disk file, most numeric variables are stored with a length of four or fewer bytes. Variable lengths were selected to accommodate the largest value occurring in the data and should not influence calculation accuracy, since the system uses double-precision arithmetic for statistical manipulations.

Special Output Formats

The SAS FORMAT procedure allows the user to define output formats, giving label values to variable values. In relation to the LRGVESS Data Base, EPA created a library of stored codes. To substitute English labels for codes, the user program must (1) include the job control language (JCL) for the cataloged format load module library, (2) construct a variable from the code variable to be used for the label, and (3) include an appropriate FORMAT statement. An alternative but slower method would be to merge files with dictionaries. Automatic procedures to accomplish this are being implemented and are discussed in this paper..

Using the Graphic User Interface under the point-and-click menus automatically associates labels with the proper data elements.

PROCESSING NEW DATA

New data sets received by LRGVESS go through a series of steps (Figure 1) involving file copying, reformatting, and QA procedures to produce "certified" data sets. Figure 2 is the worksheet used to keep track of a data set as it progresses through the steps shown in Figure 1.

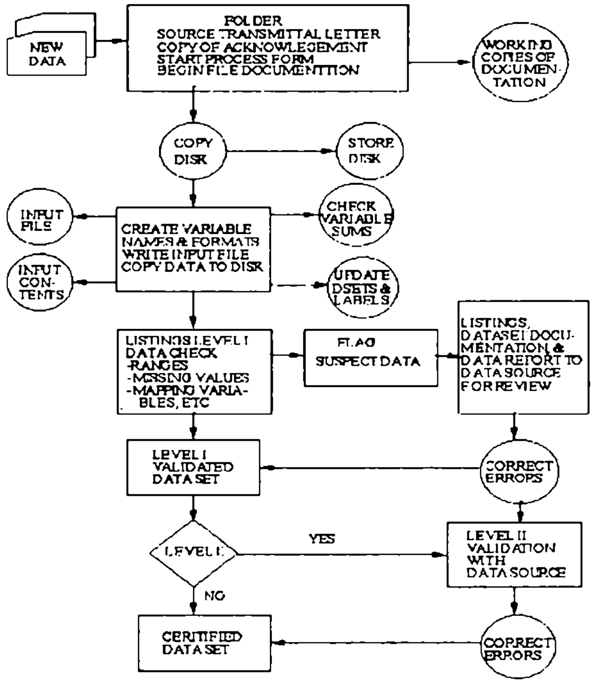
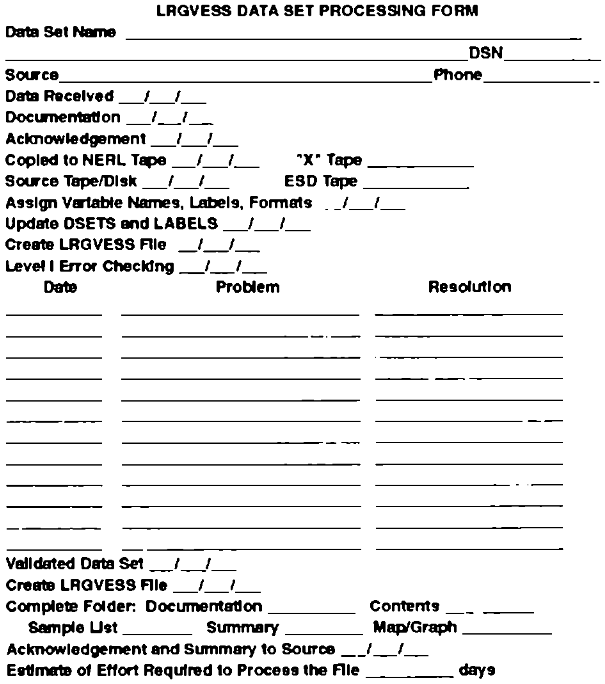


Figure 1: Certified Dataset Flow Diagram



LRGVESS DATA SET PROCESSING FORM

Data Set Name ______________________________

______________________________ DSN __________

Source ______________________________ Phone __________

Data Received ___/___/___

Documentation ___/___/___

Acknowledgement ___/___/___

Copled to NERL Tape ___/___/___ "X" Tape __________

Source Tape/Disk ___/___/___ ESD Tape __________

Assign Varlable Names, Labels, Formats ___/___/___

Update DSETS and LABELS ___/___/___

Create LRGVESS File ___/___/___

Level I Error Checking ___/___/___

Date Problem Resolution

Valldated Data Set ___/___/___

Create LRGVESS File ___/___/___

Complete Folder: Documentation __________ Contents __________

Sample List __________ Summary __________ Map/Graph __________

Acknowledgement and Summary to Source ___/___/___

Estimate of Effort Required to Process the File __________ days

Figure 2: Processing Form

Depending on the format, size, complexity, and problems encountered, the process takes from 1 to 4 weeks.

INITIAL PROCESSING

The initial processing step assigns a data set number (DSN) and creates a file folder of all documentation and processing records. The original files are copied to an UNIX hard disk that becomes the LRGVESS working copy. The original files are then stored. The documentation is reviewed, and letters are sent to the suppliers of the data acknowledging receipt of the data, indicating any problems in copying the tape, and requesting additional documentation if necessary.

The second phase of processing consists of converting the data to a SAS data set. Variable names, labels, lengths, and formats are assigned to each variable. The names and labels are not assigned based on the user-supplied documentation but are recoded to the existing LRGVESS variable names to ensure uniqueness and overall consistency. If new variables, units, compounds or elements are needed, the variable names and labels are added to the LRGVESS dictionary.

A SAS input program is written to read in the data using the assigned variable names. The program converts units of measure to metric units if necessary and generates listings of the data, frequencies of codes, and statistics to assist in error checking. The SAS "working" data set is stored on an online mass storage device with access limited to the LRGVESS staff.

A second letter, containing SAS output lists and the LRGVESS data set description, is sent to the data supplier for review. Any problems that need resolving are highlighted. Errors are then corrected on the LRGVESS file. This process may be repeated as needed to obtain the certified data set. This data set is then transferred to a disk area with less restricted access. At this point, the data is available for loading into ORACLE, through SAS/ACCESS, and for distribution to other users.

LRGVESS staff next make sure that the data set folder includes records of the intermediate and final processing steps and contains the complete documentation. Finally, the written description of the data set and the list of labels and variables are entered into the LRGVESS Notebook.

QUALITY ASSURANCE

Experience with extant data has emphasized the need for extensive data checking. The first step in data QA is to document the data source fully, noting any constraints on the interpretation of the data, collection methods, etc. Variable names are checked for uniqueness and clarity and changed if ambiguous or inconsistent. Units of measurement are converted to SI metric units if necessary.

Level I error checking uses relatively simple procedures to verify that the dataset represents what is described in the documentation. Missing data and outliers are highlighted, as are latently incorrectly coded data (e.g., grossly inaccurate compound values). Data problems encountered include erroneous data, missing data, wrong variable identifiers, and inconsistent variable identifiers for samplers, elements, compounds, units and other items. (For example, plotting compound values revealed points outside the error bands expected). The most fruitful ways of determining errors are to (1) use the data, (2) generate thematic maps and plots, (3) search sorted lists, (4) calculate univariate statistics, and (5) associate two or more independent files.

Although users of a data set may eventually discover errors, a preliminary QA analysis of a variable quickly displays possible inconsistencies through irregularities in the expected pattern.

More sophisticated and time-consuming QA methods can be used on selected data sets where necessary. These include graphical displays and statistical testing to identify suspected outliers. One useful approach is to compare new data files against older or similar files for consistency (e.g., compare updated compound statistics against earlier survey data). If potential problems are discovered with these checks, the data supplier is contacted to help resolve any errors.

Errors which are very apparent such as incorrect magnitude of unit measurements are corrected by the quality assurance personnel, noted in writing and sent to the laboratory for confirmation. Other errors which are discovered are not corrected until approved by the originating laboratory in writing. All corrections to the database are entered into a log which is part of the dataset so that analysts working with the LRGVESS data will know how the data was changed. For example, an analyst may have created a spreadsheet from some of the data and be working with it on a personal computer. Since these analysts are not actively accessing the database when the changes are made, this allows them to know that data modifications have occurred that might impact their results. It is recommended that each analyst either rerun results before submission for publication from the database or check to make sure that the data they are using has not been modified.

Another QA method, one of the most thorough for ensuring quality control, is to combine two independent files and check for inconsistencies. If a variable is found to have such an inconsistency, both data sources are checked. If an error is found, each file is checked for errors, paticularly in transcribing and entering data. Appropriate corrections are made using the edit procedures.

Suspicious data values are brought to the source researcher's attention for possible correction. Corrected data sets are then combined, divided, or summarized into logical relational sets, depending on their content and intended use. Despite continuing efforts toward quality control in the LRGVESS Data Base, it would be presumptuous to imply that the data base is error free. Errors are corrected when found. Individuals obtaining copies of the data base must share the responsibility of informing the database administrator of possible errors discovered in the data. Users should also periodically request information on updates or request an updated copy of a particular file or read the log file for the database.

DATA EXCHANGE FORMATS

Access

Access to data sets in LRGVESS can be either online using the computer system at EPA (LRGVESS staff) or, less frequently, through data provided to individual users. Files can be transferred to other computers either as ORACLE, SAS, LOTUS, or ASCII files. The ORACLE and SAS files contain all the dictionary information necessary for an installation with ORACLE or SAS to establish the data base. Formatted ASCII files require creating a format for each data set, using PROC SQL statements to write the file, and writing a computer program for the new user to read and process the data.The following conventions are used in providing data files.

Documentation

Data exchange requires that adequate documentation accompany data files, because it is unreasonable to expect that all files can be created in a uniform, standard manner. LRGVESS has established guidelines designed to promote consistency and to simplify file transfer. The contents of data bases should be described using a Data Base Description form.

Contributors are encouraged to identify additional published supporting documentation, including examples of applications. In addition to describing the data, the form identifies the appropriate reference citation to be used by secondary users of the data set in acknowledging the primary source.

File Contents

The following conventions are preferred within files: (1) SI metric units of measure, (2) missing values as ".", dates in YYMMDD format, and times in 24 hour notation

Data Security

Data security for LRGVESS data sets involves two aspects: protection from data corruption and loss and prevention of illegal access.

Protection from Data Corruption or loss

Data received from researchers are first copied. LRGVESS works only with the copy; the original diskette/tape is placed in a data storage area for future reference, if necessary. Datasets are transferred from tape to mass storage devices, which themselves are backed up daily and are kept for one year. In addition, working programs manipulating the data are kept in a disk area that is backed up daily. Corrupted or destroyed data can be easily restored from these sources.

Data security is further enhanced by password protection, data encryption, and field protection utilized in SAS, and ORACLE.

CUSTOMIZED SAS WINDOWS

Standards are critical in developing applications. It is both convenient and efficient to keep operating system files and SAS data files in separate libraries.

Besides the obvious that data files have different optimal storage factors such as block size and logical record lengths, code catalogs are more static and do not involve as many changes. Backup procedures also are different with data files needing more frequent backups. Another important standard is that generic code should be kept in a separate SAS library or frequently used SCL routines and specific code in a different library. This avoids errors of using application specific code when the user needs generic code and vice-versa.

Probably the most important standard is that catalogs, libraries, source entries, and data files have meaningful names and at the beginning of each entry a description of what is being accomplished, a list of the variables with full descriptions, and the dates and reasons for changes. This will save time in that explanations are given for the changes and why one approach may not have worked efficiently or at all! This is done by a change block at the beginning of a program module. It is much better to have more comments than less in a program module. It is recommended that changes be included in the program module rather than a separate file for ease of maintenance. An example of a module or program description is the following:

```
/********************************************

Date Created: 6/12/95

Author: Ron Matheny

Purpose: This program invokes the browse object
from the SAS object library.
********************************************/
```

A change block would be similar:

```
/********************************************

Date Changed: June 20, 1995

Person Making Change: Ron Matheny

Reason For Change: This change reads the possible
names of datasets and views from an ASCII dataset
rather than a SAS LIBREF.

Details of change: The code was modified to use a
FILEREF rather than a LIBREF for legitimate dataset
and view names. A simple macro to add the four level
SAS reference was added.
********************************************/
```

Also, to make administration of error messages and notes simpler, messages can be stored in an array which makes adding and maintaining the messages efficient. An example of a message array is:

```
/********************************************/

array messages {*} $ 65 (

'Error 001: - No Dataset or View Selected -- Select
Dataset or View'

'Error 002: - Dataset of View is empty --Select another
Dataset or View'

'Error 003: - You do not have Access to this Dataset
or View --Select Another'

'Error 004: Invalid Value --Please enter the correct
value'

'Note 001: Dataset or View is open'

'Note 002: Data has not yet completed verification'

'Note 003: You have included duplicate samples'

   );
/********************************************/
```

For ease of maintenance the maximum array size is not coded. This allows easy update of the array and changing the maximum size of the array is not needed. To assign a message to a display filed an ssignment statement is used:

```
_msg_=message{3}
```

It is possible to use SCL to create a SAS dataset containing the messages and may be preferred. This has the advantage of allowing messages to be assigned using the GETITEMC function and defining the messages only once for an application. It does require overhead in that the list must be setup at the application start. The form of the GETITEMC function is:

```
_msg_=getitemc(listid, 4);
```

TABLES IN CUSTOMIZED WINDOWS

Most analysts do not use "raw" monitoring data for their analyses. Instead, data is aggregated to some summary level to be used. Summary data for EPA's Lower Rio Grand Valley and the use of customized windows for data selection simplifies this task. The data is extracted from ORACLE through ORACLE and SAS VIEWS for database updates and general analysis. The major advantage of customized screens is that menus can be used to eliminate (or at least reduce) errors made by the database administrator or an analyst. A SAS macro is used to construct a WHERE statement and SQL code necessary to create a view using SAS/ACCESS and PROC SQL from the following screens. The system administrator can create the tables by sorting the data and using _FIRST_ and _LAST_ variables to write to either an ASCII file or SAS file as desired. Each value is extracted from the ORACLE database using SAS/ ACCESS.

The following tables are illustrative of those used in the EPA study.

Year of Study
1993
1994
1995

Residence Codes
Residence 1
Residence 2
Residence 3
Residence 4
Residence 5
Residence 6
Residence 7
Residence 8
Residence 9
Central (or fixed) site

Samplers
Microenvironmental sampler
Loe-volume sampler
Canister sampler
VAPS - central site
MES - central site
PS-1 - central site
Dichot sampler
Performance evaluation sample
Method evaluation sample
Recovery standard sample
House dust
Yard soil
24 hour mixed solid food
Aveolar
Whole blood sample
Blood serum sample
Urine - first morning void sample
24 hour composite sample

Analyte group
Metals
Acid aerosols
Pesticides
Volatile organics
Polynuclear aromatic hydrocarbons
More than one group or unspecified

Sample collection location
Primary participant
Indoor - residence
Secondary participant
Indoor - residence
Outdoor - residence

Sample type designator
Sample
Duplicate sample
Field control
Lab control

Since several hundred compounds are contained in the database, a list of compounds is also available from which the database administrator or analyst can select what is to be modified or analyzed. In addition, the user may enter a compound name, which is checked against the list. If the compound is not listed, it will return a list of compounds with similar spelling.

The user then points-and-clicks on the correct compound.

Once the database VIEW is created through OOP-SAS, additional manipulations may be performed on the data such as editing, updating, inserting new data values, graphs, maps, and time series analysis. Most of these functions can be used directly from the SAS Object library with little or no code modifications or using SCL. It is possible to use SAS/ASSIST menus or go further and develop customized windows and related objects for special analysis.

CONCLUSION

This paper has provided the skeleton of an analysis system being used by EPA. Once data is entered into ORACLE, through SAS, OOP Database managemnt tools simplify data retreival. Most of the data retrieval is done through customized tables using lists rather than using default SAS objects. Database administration is critical for sucessful analysis of data and only through careful implementation is it possible to effectively and efficiently obtain qualitdata for analysis.

REFERENCES

Koch, G., ORACLE7: The Complete Reference, Osborne McGraw-Hill: New York, 1993.

Matheny, R., "Interfacing SAS to ORACLE in the UNIX Environment", *Computing in Enviornmental Management*, Proceedings of the International Specialty Conference, Air & Waste Management Asoociation, November 30 -December 2, 1994.

Matheny, R., "Lower Rio Grande Valley Environmental Scoping Study: Data Management Guide", Draft Technical Report, United States Environmental Protection Agency, January 20, 1995.

Matheny, R., "Managing Large Databases using SAS and ORACLE", Accepted for presentation at the SouthEast SAS User's Group Conference, September 10 - 12, 1995.

SAS Institute Inc.,*Getting Started with SAS/ACCESS Software, Version 6, First Edition*, Cary, NC: SAS Institute Inc., 1994.

SAS Institute Inc., *Getting Started with the FRAME Entry: Developing Objject-oriented Applications, Version 6, First Edition*, Cary, NC: SAS Institute Inc., 1993.

SAS Institute Inc., *ORACLE7 Server SQL Language Reference Manual, First Edition*, Cary, NC: SAS Institute Inc., 1993.

SAS Institute Inc., *SAS/AF Software: Frame Entry Usage and Reference, First Edition*, Cary, NC: SAS Institute Inc., 1993.

SAS Institute Inc., *SAS/AF Software: Usage and Reference, Version 6, First Edition*, Cary, NC: SAS Institute Inc., 1989.

SAS Institute Inc., *SAS Companion for the UNIX Environment and Derivatives, Version 6, First Edition*, Cary, NC: SAS Institute Inc., 1990.

SAS Institute Inc., SAS *Guide to the SQL Procedure: Usage and Reference, Version 6, First Edition*, Cary, NC: SAS Institute Inc., 1989.

SAS Institute Inc., *SAS Language: Reference, Version 6, First Edition*, Cary, NC: SAS Institute Inc., 1990.

SAS Institute Inc., *SAS Screen Control Language: Reference, Version 6, Second Edition*, Cary, NC: SAS Institute Inc., 1994.

SAS Institute Inc., *SAS Technical Report P-222: Changes and Enhancements to Base SAS Software, Release 6.07*, Cary, NC: SAS Institute Inc., 1991.

SAS Institute Inc., *SAS Technical Report P-252: Changes and Enhancements to the SAS System, Release 6.09*, Cary, NC: SAS Institute Inc., 1994.

SAS Institute Inc., *SAS Technical Report P-254: Using the SQL Query Window, Release 6.08*, Cary, NC: SAS Institute Inc., 1994.

SAS Institute Inc., *SAS/ACCESS Interface to ORACLE: Usage and Reference, Version 6, Second Edi-*

tion, Cary, NC:SAS Institute Inc., 1993.

SAS Institute Inc., *SQL Language Reference Manual, Version 6.0*, Cary, NC: SAS Institute Inc., 1994.

Stanley, D., *Beyond the Obvious with SAS Screen Control Language*, Cary, NC: SAS Institute., 1994.

For Further Information

The author may be reached by mail at the National Exposure Research Laboratory, United States Environmental Protection Agency, MD-78B, Research Triangle Park, NC 27711.

Telephone: (919) 541-2983

FAX: (919) 541-0920

Internet Address: matheny.ron@epamail.epa.gov.

TECHNICAL REPORT DATA

(Please read Instructions on the reverse before completing)

1. REPORT NO. EPA/600/A-95/147	2.	3. RECIPIE

4. TITLE AND SUBTITLE

Managing Large DataBases with Customized SAS Windows

5. REPORT DATE

6. PERFORMING ORGANIZATION CODE

7. AUTHOR(S) Ronald W. Matheny

8. PERFORMING ORGANIZATION REPORT NO.

9. PERFORMING ORGANIZATION NAME AND ADDRESS

National Exposure Research Laboratory
Office of Research and Development
U.S. Environmental Protection Agency
Research Triangle Park, NC 27711

10. PROGRAM ELEMENT NO.

11. CONTRACT/GRANT NO.

12. SPONSORING AGENCY NAME AND ADDRESS

National Exposure Research Laboratory
Office of Research and Development
U.S. Environmental Protection Agency
Research Triangle Park, NC 27711

13. TYPE OF REPORT & PERIOD COVERED

Symposium Proceedings

14. SPONSORING AGENCY CODE EPA/600/09

15. SUPPLEMENTAL NOTES

16. ABSTRACT

This paper discusses the principles of database management through customized windows using SAS/AF, particularly PROC BUILD, to invoke interactive and batch processing of data entry, editing, updating, automatic report generation, and custom report generation functions, including graphics. SAS/ACCESS is used to efficiently store the data in ORACLE using about a third of the memory that SAS databases require. Customized "views" are developed using PROC SQL with PROC/ACCESS and BASE/SAS to efficiently manage extraction of the data from the ORACLE database into SAS for a variety of purposes. Security from both the ORACLE and SAS sides is discussed as it relates to a network of personal computers and Unix workstations and Unix servers including storage of data on different Unix machines to allow more effective controlled access to the data.

This paper also gives users a detailed skeleton of the manual, automated data entry and quality assurance procedures that may be needed for proper database management before customized windows are used.

This paper has been reviewed in accordance with the United States Environmental Protection Agency's peer and administrative review policies and approved for presentation and publication. Mention of trade names of commercial products does not constitute endorsement or recommendation for use.

17. KEY WORDS AND DOCUMENT ANALYSIS

a. DESCRIPTORS	b. IDENTIFIERS/OPEN ENDED TERMS	c. COSATI Field/Group

18. DISTRIBUTION STATEMENT Release to Public	19. SECURITY CLASS *(This Report)* Unclassfied	21. NO. OF PAGES
	20. SECURITY CLASS *(This Page)* Unclassfied	22. PRICE