Developing an Air Quality Data Management System
Patricia Tobin, Salt River Project, Phoenix, Arizona

ABSTRACT
Managers of Air Quality databases must review and validate large quantities of data. A data management system using Version 6 SAS® software has been developed at the Salt River Project, Phoenix, Arizona, to replace a system which was created in the early 1980's. The original system was used to perform all necessary functions, including data entry, uploads from data loggers, validation, and the final storage of validated data. This system was originally developed using Version 4 SAS software (later converted to Version 5) and was comprised of SAS/FSP® screens in conjunction with SAS DATA steps and procedures. It has now been replaced with a data management system in Version 6 SAS software, using SAS/AF® and SAS® Screen Control Language. Extensive use is made of features not previously available, including SUBMIT blocks, interactive data validation and computations using both SAS SCL and table lookups, and the ability to dynamically subset data for statistical (PROC MEANS), graphical (PROC GPLOT), and reporting purposes. This new system has greatly simplified validation processes, and in doing so, has significantly increased efficiency and accuracy.

INTRODUCTION
The Salt River Project (SRP), an electric and water utility based in Phoenix, Arizona, has been involved for 10 years in the collection of visibility and meteorological data in northern Arizona. Data collection had originally been done as part of a visibility study in this region, in cooperation with several government agencies and other utilities. This study has been completed, but the monitoring program is being continued by SRP.

Meteorological data collection at the SRP sites includes hourly average ambient data and daily upper air weather data from radiosonde balloons. Visibility data includes photography, filter results, light scattering data, and elemental analysis. Each of these data types requires unique collection methods, quality assurance/quality control (QA/QC) data, and validation processes.

To aid in these efforts, the on-line Air Quality Data Management System (AQDMS) has been developed using SAS/AF and SAS SCL. SAS/AF screens are used for input of primary data and secondary QA/QC information, upload of data from loggers, merging data files, performing calculations, providing graphical and statistical data summaries from interactively-defined user parameters, and for transferring data to permanent storage in SAS data libraries.

This paper will present the main features of the system, and discuss some of the functionality now available in Version 6 SAS software. Probably the most significant feature is the ability to perform data processing functions interactively. Specifically, the ability to perform table lookups, build logic behind screens, and to generate and submit data steps and procedures interactively, have greatly simplified data management processes, and increased efficiency and accuracy.

ORGANIZATION AND COMPUTING ENVIRONMENT
The Engineering department at the Navajo Generating Station (NGS) and the Air Quality division of the Environmental Services (ES) department in Phoenix, cooperatively define and implement the air quality monitoring program, as well as administer research projects associated with these data. The NGS Engineering department has primary responsibility for the data collection and initial data review. The Systems and Information Resources division of the ES department (S&IR) is responsible for providing computer support for data collection, validation, archival, and research purposes.

The SRP has a large IBM mainframe, running both VM and MVS. The AQDMS is run under MVS. The mainframe system is accessible at most SRP facilities throughout the state, including the generating stations. At this time only personnel at NGS and in Phoenix are using the AQDMS.
REPLACEMENT OF EXISTING SYSTEM

The original data management system used to process these data was written in Version 4 SAS software in the mid-1980's. It was apparent by 1990 that it was time for replacement of the existing system. There were three main reasons for this decision. First, over time, weaknesses had become apparent in the existing system. Because of these, it was often necessary to by-pass the system, perform manual computations, and edit the sequential data files directly. Second, the system had been extensively modified over the years, with a significant amount of hard code added. Third, SAS Institute Inc. has greatly enhanced the SAS/AF and SAS SCL products, now offering increased power and flexibility in developing applications. These factors drove the efforts to create a new data management system.

DATA COLLECTION AND STORAGE

A PC based data acquisition system(DAS) at NGS collects much of the air quality data. Additional data are collected on paper charts. Filter analyses are provided by laboratories. These data are input into the system using 3 methods. One, data from the DAS are uploaded to sequential files on the mainframe. The AQDMS is then used to transfer these data to SAS data sets for processing. Two, basic filter sample information is entered into SAS/AF screens through the AQDMS at NGS, and written to SAS data sets. Three, laboratory filter analyses are received on diskette, and uploaded to sequential files on the mainframe at ES. The AQDMS system is used to transfer the results to SAS data sets, and merge these with the previously entered filter information.

Besides the primary data collected, two additional types of data are necessary to complete the data processing. First, tables have been created for reference and validation. These contain site information, error flag definitions, validation limits for data parameters, and status tables to track the validation process. Second, QA/QC datasets are created containing field and analyst comments and various other information including calibration, mass, uncertainty, accuracy, and precision data. Data are loaded to these tables during the transfer processes and by data entry.

Although large quantities of data are processed through this system, the actual number of users involved in the data collection and validation is small. Responsibility for data (generally in monthly batches), is passed from one area to another as tasks are completed. For this reason, complete data and task separation is required, and no sharing of files was incorporated at the data entry and validation stages.

SYSTEM DESIGN

Although information collected for each data type is unique, care was taken to create consistent user-interfaces. Each data type uses similar menus and general screen construction, consistent PFkey definitions, field definitions, and error handling. From the programming side, although unique data requirements necessitate separate programs for each data type, a significant amount of effort was made to ensure consistency in the screen layouts and in the SAS SCL programs. The following describes the design of the filter sampler (SCISAS) subsystem, but the subsystems for the other data types is similar.

A hierarchical system of menus allows selection of data type and function. The initial menu shown in Figure 1 contains selections for each of the data types.

Figure 1:

```
AIR QUALITY DATA MANAGEMENT SYSTEM
SALT RIVER PROJECT - AIR QUALITY VALIDATION MENU
Place cursor on your choice and press enter:
GASEOUS: (PLANT EMS) (SURFACE) (GASEOUS)
METEOROLOGY: (SURFACE) (UPPER AIR)
VISIBILITY: (PHOTO) (NEVIS)
PARTICULATE: (DICHOT) (NIKOL) (GRAV)
(ELEMENTAL) (ABSORPTION)
(EXIT AQ DATA FILE MGMT)
```

Upon selecting a data type, the appropriate library is opened, and the menu shown in Figure 2 is displayed. This menu allows selection of specific functions, or transfer to the menus for reference and QAQC data. These menus are all SAS/AF screens using SAS SCL code.

Figure 2:

```
SCISAS DATA MANAGEMENT SYSTEM
Place cursor on selection and press ENTER key:
(DATA ENTRY) (TRANSFER TO VAL DS)
(UPLOAD DATA) (MERGE MASS DATA)
(VAIDS) (PRINT DATA SETS)
(STATS/GRAPH) (TRANSFER TO FER DS)
(DISPLAY QAQC) (DISPLAY REF TABLES)
(EXIT SCISAS)
```
Code for menu screens is simple, mainly consisting of calls to DISPLAY routines to display selected screens, as shown in Figure 3.

**Figure 3:**
```
MAIN:
call wdef(1,1,24,80);
select(type);
when('date') call display('scisas.program');
when('transval') call display('transval.program');
when('massmerg') call display('massmerg.program');
when('valid') call display('validat.program');
when('print') call display('printds.program');
when('sanldg') call display('sanldg.program');
when('tranper') call display('tranper.program');
when('menu2') call display('menu2.program');
when('menu3') call display('menu3.program');
when('return') _status_ = 'E';
otherwise;
call wdef(2,2,23,79);
cursor return;
type = ' '; turn;
call display('scisas.program');
call display('transval.program');
call display('massmerg.program');
call display('validat.program');
call display('printds.program');
call display('sanldg.program');
call display('tranper.program');
call display('menu2.program');
call display('menu3.program');
```

The basic functions handled for each data type, and which will be described in more detail in the remainder of the paper, are:
- data entry
- merge data from sequential files
- validation of data
- print datasets, statistics, or graphics
- transfer to archive
- browse and edit QA/QC data sets and reference tables

**DATA ENTRY:** As data are entered, validations, calculations, and error checking are performed. Verifications are made against the reference tables. Calculations in SAS/SCL are performed as data is entered, with results immediately displayed. Error checking is performed in several ways: screen variable attribute definitions, SAS/SCL cross validations, and validations against QA/QC data sets and reference tables. Subroutines are provided to allow search for and fetch of specific records based on key information. Reference tables are accessible through the HELP function, enabling the user to authenticate data entry without closing the current screen. This is accomplished by specifying the Reference Tables menu program in the GATTR screen as the general help screen.

**TRANSFER TO VALIDATION:** Once a batch of data has been entered or uploaded from a sequential file, the data manager transfers the data to a validation data set. The SAS/AF Transfer screen in Figure 4 enables the user to either transfer the entire contents of the data set, or to specify a subset based on sites and timeframes.

**Figure 4:**
```
SCIAS TRANSFER TO VALIDATION DATA SET
To transfer all sites for specified date range, enter date range:
DATE RANGE: __ __ __ __ (mm/dd)
```

Several tasks are completed as part of the transfer process. First, data from the QA/QC data sets and reference tables are merged. Second, calculations are performed, incorporating values from the calibration table. Last, outlier flags are set based on values from the limits table. The actual data transfer is performed by using a SUBMIT block to submit DATA steps, with the 'continue' parameter to allow for immediate execution. The site and date information entered on the screen are incorporated into the code as 'if statements' to allow subsetting of the data. This functionality is discussed in detail in the section on graphics and statistics reporting.

**UPLOAD:** The upload menu option is used to upload data received on diskette. A SAS/AF screen allows the user to enter the name of the sequential data set to be uploaded. This data set is allocated and the SUBMIT block in Figure 5 is executed to upload the data.

**Figure 5:**
```
SUBMIT CONTINUE:
TWO ALLOC ENDDATA DISCARD NAME='SHI';
Data scisas.mess;
input date:
length flexid 52
  weight 4;
flexid flexid 52.
  weight 4;
label flexid = 'FILTERID' weight = 'WEIGHT';
input @1 flexid 5. Q;
qy = weight/flexid;
if subqr(qy,1.1) & 'A' or subqr(qy,1.1) = 'E'
  then delete;
else do;
qy @14 weight 4;
  output;
end;
END; SUBMIT:
```
MERGE: The merge option merges the data just uploaded from diskette with the previously entered sample data. The SAS/AF screen allows the user to enter required date information and request the merge. The entire merge is executed from within a SUBMIT block containing SAS DATA steps. These DATA steps merge the two and then merge the results with the limits table to set outlier flags. Reports are printed giving the status of the merge, detailing exceptions encountered. This provides the user with the necessary feedback as to any problems with the data.

VALIDATION: The validation screens are generally similar to the data entry screens, with the addition of calculated parameters and error flags generated during the merge and transfer processes. The SCISAS validation screen is shown in Figure 6.

Figure 6:

Figure 7:

```
* VERIFY CALIBRATION DATE *

VERCALIB:
  where cl = cl into cl;i=1;
  if (cl = -1) then do;
    msg = "Calibration date is not found in the calibration data set";
    error notice;
    cl = data("not found");
    exit;
  end;
  else do;
    cl = data(calid,cl;i=calid,cal);
    adjust = geterr(cal);
    all = waserr(all,cal);
    if (adjust > 0) then do;
      msg = "Calibration date found in the calibration data set";
      error notice;
      cl = data(cl,cal);
      exit;
      all = waserr(all,cal);
    end;
    reset;
  end;
```

Since the original data will be preserved, additional fields are provided for entering data modifications. When modifications are made, the error checking module will validate against the QA/QC data sets and reference tables, and the computation module will be executed to recalculate computed fields. The WHERE and FETCH functions are used to select the support information, and the GETVAR function in Figure 7 is used to retrieve values.

Calculated fields are automatically updated whenever any associated parameters are updated. The user may only enter error codes in calculated fields to prevent inconsistencies. These are significant improvements over the original system, which forced the user to use extensive manual procedures to recalculate data.

TRANSFER TO ARCHIVE: When validation is complete, data may be transferred to the archive library. The screen used to initiate this process is similar to the transfer to validation screen. The user may also specify a subset of the data to be transferred. Once the user enters the subsetting information, SAS DATA steps are immediately executed to perform the transfer. DATA steps are submitted using a SUBMIT block with the 'continue' parameter to allow for immediate execution.

The following sections describe several functions which are common to many of the screens. For this reason, these functions were not reviewed above.

DYNAMIC GRAPHICS FROM VALIDATION SCREENS
Since one of the main goals in designing this system was to increase accuracy and efficiency, additional features were added where deemed helpful to the data validators. One way in which this was accomplished was to provide a PFkey on many validation screens to allow the user to instantaneously view a graphics display of the data being reviewed. This provides an efficient way to spot outlier data and trends.
USE OF EXTENDED TABLES

The QA/QC data sets and several of the reference tables are displayed using extended tables. The use of extended tables, as in Figure 8, was incorporated whenever it was more efficient for the user to input and process multiple records at a time.

Figure 8:

<table>
<thead>
<tr>
<th>SCISAS LIMITS SCREEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>PF1=HELP PF3=ADD PF6=TOG PF7=BACK PF8=FORM PF12=END</td>
</tr>
<tr>
<td>SITE NAME</td>
</tr>
<tr>
<td>DIFF MIN MAX</td>
</tr>
<tr>
<td>BRCA F_PRESS 0.5</td>
</tr>
<tr>
<td>BRCA T_PRESS 0.5</td>
</tr>
<tr>
<td>BRCA F_FLOW 90 10/17 124.3</td>
</tr>
<tr>
<td>BRCA T_FLOW 90 3</td>
</tr>
</tbody>
</table>

Note that the screen consists of a top nonscrollable section, and a bottom scrollable section. The top section contains an ADD line, consisting of columns in which to add data for new records. The bottom portion of the screen scrolls through the data set. Deletes are performed by entering the code DEL in the first column of the row to be deleted. Rows flagged for delete, are actually deleted in the TERM section of the SAS SCL by executing a DATA step in a SUBMIT block. Updates are dynamically performed whenever columns are modified in the scrollable portion of the screen.

TOGGLE CAPABILITY

Most of the screens have a toggle function, which allows the user to toggle to other screens within the same data type. This is accomplished by using the SAS SCL DISPLAY routine in Figure 9 to display the main menu.

Figure 9:

```
MAIN:
* menu entry flags;
except_all;
* get function back pt key and execute;
  help = switch(1,1);
  if help = "help" then call display(mainmenu program);
  else if help = "log" then log marker;
  else if help = "update" then call update;
  else if help = "del" then call del;
  else if help = "log" then log marker;
  else if help = "log" then call log;
  else if help = "help" then help marker;
  else if help = "log" then log marker;
  call write(1,12,70);
  call display(main menu program);
  call write(2,24,43);
  end;
  else if help = "end" then call mainmenu.end;
  else log marker;
  return;
```

This functionality is extremely useful. For instance, a user may be performing data validation and discover that certain support data were incorrect or missing. The user could toggle to the support table, update the appropriate data, and return to the original screen to complete the data validation - all without having to close the validation screen.

DYNAMIC DATA SELECTION CAPABILITIES

The graphics/statistics, data transfer, and print programs incorporate some rather interesting capabilities of SAS SCL in order to allow the user to subset the data by site and time frame and to add screen information to titles. The graphics/statistics and print options also allow the user to specify the variables to be included. The user may enter the variables, or select from a list generated by the SAS SCL VARLIST function. The graphics/statistics screen is shown in Figure 10.

Figure 10:

```
SCISAS STATISTICS AND GRAPHICS SCREEN
Enter data range, or leave blank for all
BEGIN DATE (mm/dd/yy)
END DATE (mm/dd/yy)
Enter site, or leave blank for all
Enter up to 7 variables to statistics (separated by blank)
Off leave blank for all
Off enter X for a list of variables to select; leave blank
```

These programs each have a SELSUB module to create a subset of the data. The REPLACE function substitutes a replacement string for a reference to an SAS/SCL variable in the SUBMIT block. This functionality allows the screen variables to be incorporated into the code by the dynamic construction of the 'if' statement to subset the data. The source code is then generated and submitted to the SAS system for execution. The 'continue' option is used with the SUBMIT block so that execution takes place immediately. As an example, the SUBMIT block of Figure 11 shows the code in the SAS SCL program.
Figure 11:

```
SETSEL;
   replace word 'to data set is added than data';
   replace word 'add data to data set then data';
   last = 1;
   replace '1' with 'DAT1';
   replace '2' with 'DAT2';
   replace '3' with 'DAT1';
   replace '4' with 'DAT2';
   replace '5' with 'DAT1';
   replace '6' with 'DAT2';
   replace '7' with 'DAT1';
   replace '8' with 'DAT2';
   replace last with 'DAT1';
  move continue;
DATA RPLLIST;
  input GROUP $ 1-2;
  run;

data set;
   if measure $ = 'DAT1' then data;
   if measure $ = 'DAT2' then data;
   if measure $ = 'DAT1' then data;
   if measure $ = 'DAT2' then data;
   if measure $ = 'DAT1' then data;
   if measure $ = 'DAT2' then data;
end;
```

Figure 12 shows the expanded SUBMIT block with the replacement strings. This is the actual code that is submitted to the SAS system for execution:

**Figure 12:**

```
TSO ALLOC GDOT1 PRIV1 PRIV2 PRIV3;
PROC PRINT: NEW;
  OPTIONS DEVICE = GDOPMPS VPOS=32 HPOS=79 NOSTMP;
  RUN;
```

Once the data are subsetted, either the statistics or graphics module is executed. The graphics module in Figure 13 simply plots the subsetted data set. Figure 14 shows the expanded submit block form the graphics module, and Figure 15 shows the final plot.

**Figure 13:**

```
GRAPHIS:
   control select;
   * are primary dendrimers;
   link SELPRT;
   * subset data;
   set SELKRT;
   * are graphs;
   submit continue;
   AXES LABEL=$START DATE$;
   AXES LABEL=$CONCENTRATION$;
PROC COPLOT DATA=RPLLIST;
  TITLE C=BLUE 'SCISAS VALIDATION DATASET';
  TITLE C=BLUE *RED *RED;
  TITLE C=GREEN 'FINE CONC = GREEN';
  C=RED *CONC *RED;
  PLOT C=CONC $CONC$ $CONC$ $CONC$ $CONC$ $CONC$ $CONC$ $CONC$ $CONC$ $CONC$ $CONC$;
  BY SITE;
  FORMAT $CONC$ mmol/L;
  SYMBOL V=CIRCLE C=GREEN;
  SYMBOLS V=STAR C=RED;
  RUN;
```

The statistics module uses the REPLACE function to construct the VAR clause in the MEANS procedure, then executes the MEANS procedure, shown in Figure 16.

**Figure 15:**

```
SCISAS VALIDATION DATASET
FINE CONC 180.0 * 180.0 CONC = RED
CONC = RED
```

**Figure 16:**

```
STATIS:
   control select;
   * are primary dendrimers;
   link SELPRT;
   * create subset of data to be printed;
   link SLEPRINT;
   * data subset;
   replace var 'var done';
   submit continue;
   control select;
   SYSTEM SELECT * ;
   title 'SCISAS VALIDATION DATASET';
   title 2 = 'Data to be';
   proc means b=10 data=sasdata mean std min mea;
   by file;
   done;
   run;
```
The PRINT option allows the user to print pre-formatted reports, select variables from the list generated by the VARLIST function, or enter variable names to be used. If the user specifies variables, the list is passed to the SUBMIT block as a replacement string. The process is similar to those described above and is not repeated here.

CONCLUSIONS
The development of this AQDMS has been quite successful in eliminating the problems of the original system. First, the weaknesses in the design of the original system were eliminated by the redesign of the data structures and processes. Second, the hard code was eliminated by redesigning the processes and adding the reference tables. Third, and most important, the ability to interactively perform data management functions has added greatly to the flexibility of the system and positively impacted the accuracy of the data and efficiency of the processes.

FUTURE DEVELOPMENTS
This system has been developed for data processing and validation. Future plans are to develop a query system to permit other users to easily view the finalized data sets. The query system will allow users to view the SAS data sets, perform graphics and statistics, print, and generate sequential files from the archived data. The users will be able to dynamically subset data by parameters and by limiting variables selected.

FOR MORE INFORMATION
For more information, please contact

Patricia Tobin
Salt River Project
Systems and Information Resources / PAB354
P. O. Box 52025
Phoenix, Arizona 85072-2025

Phone (602) 236-5493

SAS, SAS/AF, and SAS/IFSP are registered trademarks of SAS Institute, Inc. in the USA and other countries. * indicates USA registration.

Other brands and product names are registered trademarks or trademarks of their respective companies.