# QUERYING WITH SAS® SCREEN CONTROL LANGUAGE

## A DATABASE BROWSING AND EXTRACTION APPLICATION

*MARIA K. WILLS, FIDIA PHARMACEUTICAL CORPORATION*
*ANNE ASHER, ARC PROFESSIONAL SERVICES GROUP*

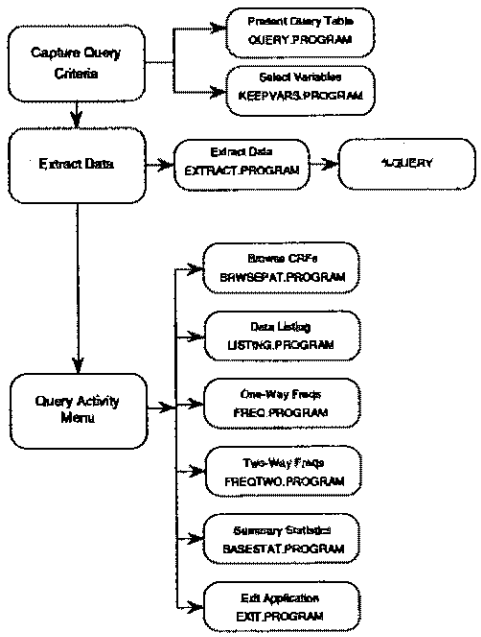
### INTRODUCTION

This interactive poster presents a database query and extraction application which was developed using the SAS Software System for Personal Computers. Version 6.03 of the SAS/AF® and SAS/FSP® products provide us with Screen Control Language, a powerful tool for developing intelligent user interfaces for interactive full-screen applications. The goal of this application is to provide the novice user with an easy method of accessing, extracting and manipulating data stored in SAS datasets without possessing knowledge of the SAS language.

### DATA LIBRARY

The data used in this application consists of clinical trials data collected using Case Report Forms (CRFs). For the purposes of this application, seven forms were chosen and separate SAS datasets created. In addition to the form specific datasets, two other datasets, the variable dictionary and dataset dictionary, are used. The variable dictionary contains the names and attributes of all variables used in the form specific datasets and the dataset dictionary contains the name of each dataset and its primary and secondary keys. These dictionary datasets are used to assist in identifying the form-specific datasets and their variables, thereby alleviating the need for the user to know how the data is stored.

*FIGURE 1. PROGRAM STRUCTURE AND CALLING SEQUENCE*




### PROGRAM STRUCTURE

Figure 1 displays the structure of this application and the calling sequence. The major component of this application is the Query and Extraction. The method used is based on Query-By-Example, a technique that is used in many SQL (Structured Query Language) applications. A fill in the blank table (Figure 2.1) is presented to the user for the purpose of capturing the criteria to matched in the data extraction step. To aid the user in completing this table, a pop-up window (Figure 2.2) can be opened which will display a list of variables available. The user can then select the desired variable from the list to be placed into the query table (Figure 2.3).

*FIGURE 2.1 BLANK QUERY TABLE*

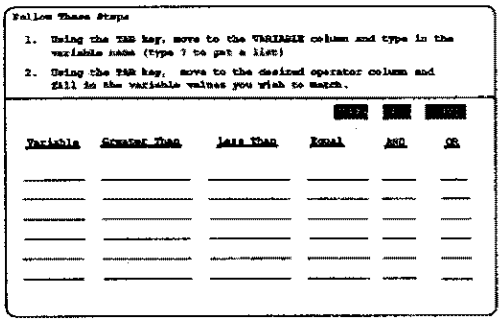



*FIGURE 2.2 VARIABLE-LIST WINDOW*

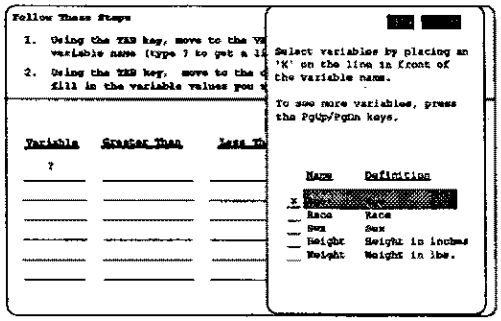
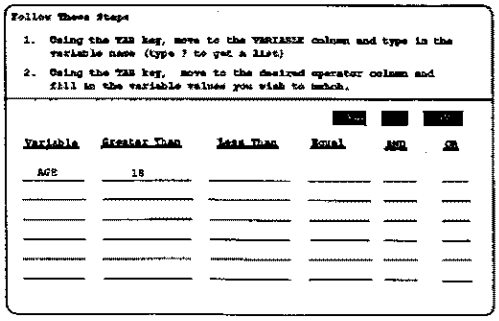



*FIGURE 2.3 COMPLETED QUERY TABLE*



Once the table is completed and submitted for processing, Screen Control Language is used to validate each entry against the variable dictionary to detect such errors as invalid variable names or invalid variable comparisons (i.e., comparing a numeric variable to a character value). Figure 3 displays a section of the SCL code used to perform variable name validation. After all fields are validated, the subsetting IF string is constructed and stored in a macro variable.

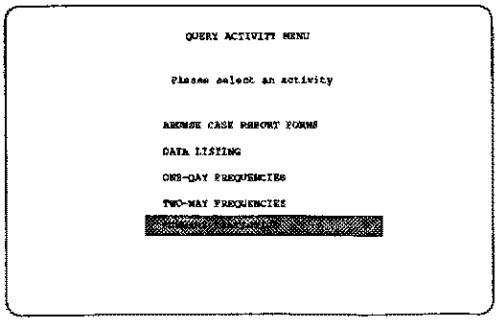
*FIGURE 3. SCL VALIDATION CODE*

```
  /*  FOR EACH VARIABLE ENTERED IN THE QUERY TABLE, CHECK */
  /*  ITS EXISTENCE IN THE VARIABLE DICTIONARY.  IF THE   */
  /*  VARIABLE IS NOT FOUND, ISSUE AN ERROR MESSAGE AND   */
  /*  RETURN TO THE SCREEN.                               */

OBSLIST = ' ';
STRING = ' ';
DSID = OPEN('DICT.VLIST','I');

DO CT = 1 TO 6;
   IF VAR[CT] NE  BLANK  THEN DO;
      OBSNO = LOCATEC(DSID,VARNUM(DSID,'NAME'),VAR[CT],'A',' ');
      IF OBSNO > 0 THEN DO;
         CH_TYPE [CT] = GETVARN(DSID,VARNUM(DSID,'TYPE'));
         OBSLIST = LEFT(OBSLIST) || ' ' || PUT(OBSNO,3.);
      END;
      ELSE IF OBSNO = 0 THEN DO;
         MSG  = 'VARIABLE NOT FOUND IN DICTIONARY';
         IF CT = 1 THEN ERRORON VAR1;
            ELSE IF CT = 2 THEN ERRORON VAR2;
            ELSE IF CT = 3 THEN ERRORON VAR3;
            ELSE IF CT = 4 THEN ERRORON VAR4;
            ELSE IF CT = 5 THEN ERRORON VAR5;
            ELSE IF CT = 6 THEN ERRORON VAR6;
         LEAVE;
      END;
END;
```

The next step in processing the query is to extract the data from the case report form datasets. This involves determining which form-specific datasets are to be used and the primary merge keys for these forms. Both the variable dictionary and the dataset dictionary are used for this purpose. Once the necessary information is gathered and stored in macro variables using Screen Control Language, the Query macro is invoked to perform the data extraction by merging the necessary datasets and applying the subsetting IF statement. If the extract was successful, the total number of records extracted is displayed to the user.

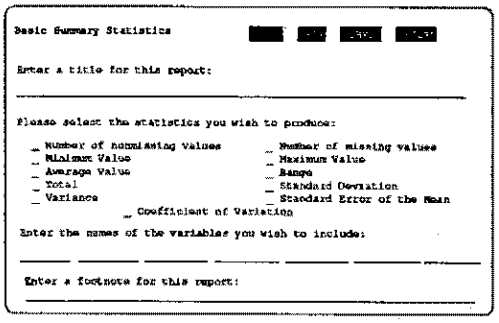
The Query Activity Menu (Figure 4) allows the user to manipulate the data extracted from the SAS datasets. Through this menu, the user can browse images of the Case Report Forms for the patients meeting the query criteria, produce reports, or perform basic summary statistics.

*FIGURE 4. QUERY ACTIVITY SELECTION MENU*




In order to produce reports (data listings, frequencies, crosstabulations and summary statistics), a fill in the blank screen is presented where the user types in titles, footnotes and variable names. Once again, a pop-up window can be opened which will display a list of available variables. Where options are available, a checklist is provided for the user to select the desired options (Figure 5). The completed screen is validated and the appropriate SAS statements are generated using Screen Control Language. If no errors are detected, the generated SAS statements are submitted to the SAS system for execution and the results displayed on the users screen.

*FIGURE 5. SUMMARY STATISTICS REQUEST SCREEN*




At the completion of each report, the user is returned to the report screen where he may generate another version of the same report or return to the Query Activity menu from which he may exit the application.

## CONCLUSION

In using Screen Control Language, the applications developer can prepare intelligent user interfaces when constructing systems for the non-SAS user. Screen Control Language can be used to perform database lookups as well as inter- and intra-field validation. The various functions available in Screen Control Language will allow the SAS Software System to be used in a wide variety of subject areas and applications.

The authors may be reached at:

Maria K. Wills, Manager EDP
Fidia Pharmaceutical Corporation
1775 K Street, N.W. Suite 800
Washington, DC 20006

Anne Asher, Senior Project Staff
ARC Professional Services Group
601 Indiana Avenue, N.W. Suite 510
Washington, DC 20004