BUILD YOUR OWN QUERY LANGUAGE
David Burrage
Department of Math and Computer Science
University of Quebec at Montreal

Introduction

MACRO statements with statement style invocations provide an elegant means of constructing query language commands. If you do not possess a data base query language for SAS (two query languages are currently available, DS/QL [1] and RAQL [2]), perhaps you should consider building your own with the SAS82 macro language.

Query language programming

The aim of a query language is to provide an easy-to-use high level interface between the casual user or programmer and SAS. Commands may be designed to have English-like syntax.

The program which follows is a sample query language program which uses macros CREATE, SELECT, PROJECT and PRINTDS to create a temporary data set and finally to print, with title, two columns of selected observations with duplicates eliminated. CREATE is a command which requires the user to specify his input data file. e.g. CMS FILEDEF etc.

CREATE TEMPDS USING "VAR1 $ VAR2 VAR3 VAR4 $" ;
SELECT TEMPDS WHERE "VAR2 = VALUE" => TEMPDS1 ;
PROJECT TEMPDS1 OVER VAR1,VAR3 => TEMPDS2 ;
PRINTDS TEMPDS2 "NEW DATA SET" ;

These are simply statement style macro invocations which provide input strings to the macros. (The double quotes surround certain input strings to allow special characters such as BLANK or SINGLE QUOTE).

These commands may be embedded naturally in any SAS program because they invoke macros which generate FROC or DATA steps. Programs containing a mixture of query language commands and normal SAS code are possible and often desirable.

Macro definitions of PRINTDS and PROJECT

%MACRO PRINTDS(DSIN,TITL) /STMT;
%*-------------------------*;
%* A macro to generate code to print a data *;
%* set with title. *;
%*-------------------------*;
PROC PRINT IF NOT (&DSIN") THEN DATA=&DSIN ;
%STR(1);
%IF &TITL " THEN %LET TITL = &SYSDSN ;
%END
TITLE &TITL ;
RUN;
%MEND PRINTDS;

%MACRO PROJECT(DSIN,DUM1,VARLIST,DUM2,DSOUT,UNIQUE=NO)/STMT;
%*------------------------*;
%* A macro to generate code to project out *;
%* columns of a SAS data set with the option *;
%* of eliminating duplicates. *;
%* COMMABL is a macro which replaces commas *;
%* with blanks in a character string to *;
%* generate a new string NEWTEXT. *;
%* SORDS() is a name-style invocation . *;
%*------------------------*;
%IF UNIQUE THEN %SORTDS(DSIN,ONKEYS,VARLIST,GIVING,DSOUT) ;
%LET VARLIST=NEWTEXT ;
%LET DSOUT=DSOUT ;
SET %IF UNIQUE THEN &DSOUT ;
%ELSE &DSIN ;
%STR(1);
%END
RUN;
%MEND PROJECT;

Note the use of dummy parameters DUM1,DUM2 in the MACRO statement. These parameters are not used in the MACRO definition but provide a means of allowing 'padding' words in the macro invocation i.e. the query language command.

e.g. DUM1 contains the string OVER and DUM2 contains the string =)

Clearly any strings could have been chosen for DUM1 and DUM2 to make the command more meaningful.

Common query language functions

A complete query language could contain several commands for each of the following categories:

- Data set creation
- Retrieval of sets of observations
- Utilities such as PRINTDS, SORTDS etc.
- Updating facilities [3]
- User HELP facilities
- Arithmetic and statistical facilities
- Graphical facilities
- Description of the tables and variables
SAS diagnostics provide some help when errors occur. However, there are types of error for which your macros should provide their own diagnostics. Henderson [4] has some suggestions and techniques.

Conclusion

When you build your own system, you choose the commands and syntax, the latter being subject to certain minor constraints of the syntax of the statement style macro invocation. Programming effort is required to establish the macros, but there will be long-term rewards in user and programmer productivity because of the higher level of programming that the use of a query language implies.

References

1. Ingram & Rothrock, "A general approach to Decision Support Systems", SUGI '84 Proceedings

2. Burrage & Gilman, "RAQL-An evolution in SAS Data Management", SUGI '83 Proceedings


4. Henderson & Kuhn, "Statement style macros", SUGI '84 Proceedings