Coding a User Application into the Executive Information System
of SAS/ASSIST® Software

Linda P. Atkinson
Economic Research Service, USDA

Introduction

SAS/ASSIST® software is a user-friendly "push-button" front end to the SAS® System. It enables users who may not have knowledge of SAS syntax to run applications that have been developed for them. The user interface to ASSIST begins with a "Primary Menu" offering as choices many of the common tasks that one would like to perform, such as Data Management, Report Writing, Graphics, and so on. One simply "clicks" on the appropriate choice (with a mouse) or moves the cursor to the field and presses Enter, to have another menu with options related to that choice presented. Submenus are brought up as needed until the task is completely specified and run, all without the end user having to write any SAS code.

A new (with Release 6.06 of the SAS System) choice on the Primary Menu of ASSIST is labelled EIS, for Executive Information System. It provides access to Public Applications provided by SAS Institute which include such Desktop Applications as Letter Applications, Calculator and Appointment Calendar. It also provides access to Private Applications developed at a site for its end users, and it includes "Build EIS", a menu-driven system which makes it easier for application developers to create push-button applications.

Documentation

There is not a "manual" as such providing instructions for using the Executive Information System, primarily because it is intended to be self-explanatory. Extensive help screens throughout EIS give information on use of the system. For published material, one page in a booklet about SAS/ASSIST [3] introduces EIS. A paper presented at the 1990 NorthEast SAS Users Group Conference [2] provides further description and an example. A technical support document from SAS Institute gives more technical details [4].

Example: A Forecast Evaluation System

The particular example used for this tutorial is a Forecast Evaluation System (EVAL), previously built as a SAS/AF® application called from the Applications push-button on the Release 6.03 SAS/ASSIST menu [1]. EVAL was developed at the Economic Research Service in response to researchers' requests for a comprehensive, easy to use package for evaluating price forecasts. Written in Screen Control Language (SCL) using PROC BUILD in SAS/AF software, EVAL provides the user with a screen on which to specify analysis options such as the data set to be accessed, the variable of "actual values" to be used, the variables of "forecast values" to be compared to actual values, the time periods to be looked at, and various statistics to be calculated (statistics of fit, trend statistics, and so on). Depending on the analysis options chosen, the application generates code using procedures in SAS/STAT®, SAS/ETS®, and/or code written with SAS/IML® software. The Screen Control Language used in the AF application also allows field validation, checking, for example, that more than one forecast variable is specified if the user has requested that revision ratios be calculated.

By incorporating the AF application within the SAS/ASSIST framework, the user also has available all the other menus for commonly performed tasks within the SAS system. If graphics or tabular displays are wanted, these can be obtained by making appropriate choices from the ASSIST Primary Menu before (or after) running the application.
The EIS Builder

The EIS Builder uses menus to aid the applications developer in building customized menu systems. These systems can contain entries which execute SAS programs, edit SAS data sets, or perform several other functions. This tutorial will use as an example a system which accesses a previously-developed SAS/AF application. This will show how to use the EIS Builder to incorporate an AF application into the SAS/ASSIST framework, where it is easy for end users to run the application as they do other SAS processing.

The "Build EIS" option of the Executive Information System leads you to three choices: Create/update applications, Run applications, and Setup. If it is the first time "Create/update applications" is selected, SAS creates an application database called SASUSER.SASAPPL. The application database to be used can be changed by pressing ENTER on the button labelled "Application database" and choosing another one or creating a new one.

To add an application to the application database, push the "Add" button at the bottom of the screen. The types of applications which can be created are listed on the screen. The application which I am using as an example will begin with a "Main Menu" giving the user a choice of editing a data set, specifying details for the analysis to be performed (the AF application), or exiting the application. The first of these is set up by choosing "Edit/Browse a SAS Data Set" as a new application to be added.

Another panel comes up for the application builder to specify details for the data editing application. I called the application
"EDITDATA", gave it a description, and specified that I wanted edit access to a data set in tabular form. The name of the data set, PRICES, is stored in the same EVAL library as my AF application's catalog. Pressing "OK" finishes specifying the application and returns to the list of application types.

Next is the definition of the AF application that I wish this application to invoke. This is created by choosing "Display a SAS/AF Application" from the "Add a New Application" list. A panel is presented on which to fill in another application name and description, followed by the names of the library and catalog in which the AF application is stored (this was previously created using PROC BUILD in SAS/AF software).

Now that both "sub-applications" are specified (the data editing portion and the AF application), I am ready to build a menu for the system which will be the user's interface with the overall application. The EIS Builder makes it easy to set up a "Block Menu" consisting of push-button-type blocks, each of which when selected will cause SAS to perform some task, as specified by the application developer. I give the menu a name and a description, which will appear in the title bar of the application. The "block title" will appear in a bar over the blocks which contain the application options. I requested to have icons for the menu items. A color scheme may also be selected. If you choose this, you can cycle through various combinations of background and foreground colors (by pressing on the NEXT and PREVIOUS arrows) until you have the arrangement you want for your menu.

The "Edit menu text" push-button is where you define each of the entries for the block menu. Block number 1 for my application is for editing the PRICES data set. This is where I tell SAS to run the EDITDATA application when this option is selected. I specify "Edit/Browse a SAS data set" as the Type of Application, and, by clicking on "Icon", get a series of Icons to choose from. When I find the "EDIT" icon that I want to use I select it and press "OK". This becomes the icon for that menu item.

The second block on the menu is to point to my AF application, where the forecast evaluation analysis specifications are made. I indicate that FCSTEVAL is the application to be run when that option is selected, having previously defined it in a SAS/AF application.
and told SAS the catalog and library where the application is stored.

Finally, I define an exit block to be the third item on the menu. When this option is chosen the menu will terminate. All I have to do is indicate that this is an exit block, and select an icon for it.

At this point everything is defined for the application, and choosing OK a couple of times and GOBACK several more backs me out to the Build EIS menu to run it. When "Run applications" is selected a list is presented of all applications that are contained in the Application database. In this case there are two "sub-applications" contained as elements of the

"MAINMENU" which when selected presents the block menu that was defined.
When the "Edit data set" option is selected from this menu, the data set is opened for editing using SAS/FSP* Software's full-screen FSVIEW procedure. Note that the user (or even the developer for that matter) could be unaware of the program used to access the data set, since this was not specified during set-up but rather taken care of automatically by the EIS Builder.

The "Analysis Specification" phase of EVAL is an AF application which was patterned after other SAS/ASSIST programs, providing capabilities for selecting a data set, bringing up variable lists from which to make selections for analysis, etcetera. The analysis options such as whether to print the data set, calculate Thiel statistics, and so on, are YES/NO switches which change from one to the other by clicking on them (defaulting to YES if the user does not elect to make changes). Choosing RUN causes SAS code to be generated and submitted for execution, ultimately presenting the user with the output window. After viewing the output, closing the window (selecting END from the FILE pull-down) causes return to the analysis specification window of the AF application, where other selections can be made and submitted.

At this point the SAS/AF application has been successfully incorporated within the SAS/ASSIST framework with an easy-to-use menu system through the assistance of the EIS Builder. There are additional customizations that can further simplify the running of the application by the end user.

Note that when "Run private applications" had been selected the user had to make the correct choice to get to the Main Menu of the EVAL application. This can be simplified by defining the MAINMENU application as the primary private application. The Setup option on the Build EIS menu gives access to file
Among other selections that can be made in Setup, the file management options allow you to define any "librefs" needed for accessing the catalog in which the application and associated data bases are stored. For example, "Assign a new libref" under SAS Data Libraries provides the facility for specifying where data sets are stored physically. The libref so defined stays in effect until the SAS session is ended. Since this would need to be done again the next time SAS is run, I find it easier to issue the libref in an AUTOEXEC.SAS file.

Although the application has been completely defined, changes can be made to it easily through the Create/Update Applications option of the EIS Builder. For example, it would be useful to add HELP to the Main Menu. I will first use PROC BUILD in SAS/AF software to create the HELP screen. The command edit menuhelp.cbl on the catalog directory display provides an editing screen on which to type the "help" text. I prefer not to have a command line on this screen so I specify NONE on the Banner option of the Global Attributes for the help screen. After saving the entry MENUHELP.CBT to my EVALSYS catalog I return to the EIS Builder to "Add a New Application" to issue the
SAS Command "Help". This shows up as another entry in the Application database, with a type of SASCMD. Next I edit the MAINMENU application to specify as Global Help the screen that is to be displayed when the Help command is issued (EVALEVALSYS.MENUHELP.CBT).

Finally, I "Edit menu text" to add a button on the menu for calling the application which issues the Help command. To insert this new button before the last (the Exit button), I went to Block number 4 and copied the information from Block 3 into it. Then I can edit Block 3, specify that it be labelled Help and call the HELP application (a SAS command), and choose an appropriate icon. When the application is run it now comes up with a Help option to provide the user with informational assistance.
Summary

The EIS Builder in SAS/ASSIST software makes it easy for an applications developer to build a customized menuing system for site-specific applications. Not only is the end user relieved of the burden of issuing code and knowing which software products are to be used, but so is the developer, as in the automatic use of PROC FSEDIT to perform the function of editing a SAS data set. Quick tie-ins can be made to already existing SAS/AF applications as demonstrated in this tutorial. Throughout, one has access to the full power of the SAS System through the other functions of SAS/ASSIST.

References


