

Mouse Clicking Your Way Viewing and Manipulating Data with Version 8 of the SAS® System

Terry Fain, RAND, Santa Monica, California
Cyndie Gareleck, RAND, Santa Monica, California

ABSTRACT

Version 8 of the SAS® System offers several powerful ways to view and interactively manipulate data. This paper will focus on four such methods: PROC FSEDIT, PROC FSVIEW, Viewtable, and the Analyst Application. We will examine the strengths and limitations of each approach, with an emphasis on their most powerful features and ease of use. While the examples illustrate a Microsoft Windows 2000®, the concepts are equally applicable to other operating environments.

OVERVIEW

In Version 8 the ability to interactively view and manipulate data is available from within base SAS Software through PROC FSVIEW¹, the Viewtable Window, and, the Analyst Procedure. PROC FSEDIT, a procedure within SAS/FSP, is also available for interactively working with data.

One area we will cover in the paper is creating SAS data tables. To save a table you need to first define a libname and since the point of this paper is to show how to “mouse your way” through SAS, we decided to start the paper with a discussion of how to define a SAS library without using code.

LIBNAME WINDOW

While libnames are usually assigned using code, you can easily assign them using the Libname Window.

To define a new libref you need to first bring up the Libname Window by typing ‘libname’ in the command line. An example is shown below in Figure 1.

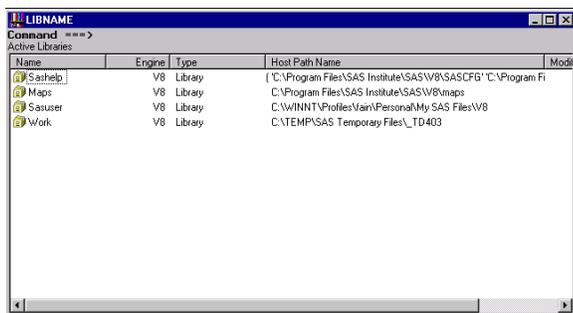


Fig. 1 - Libname Window

Click on the Libname Window to make it the active and right click over the Window to bring up the Libname Window menu. Select New. This is illustrated in Figure 2.

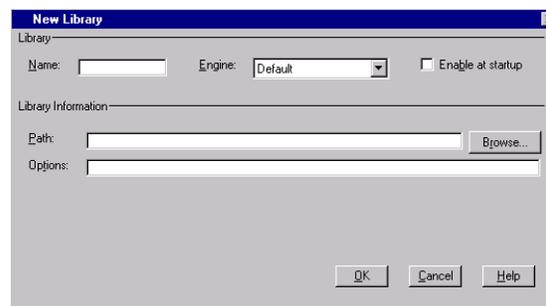


Fig. 2 - Libname Window Dialog Box

In the dialog box enter the required information, i.e., name, engine, and location. If you want this libname defined each time you invoke SAS, select the option, Enable at startup. Keep in mind that if you upgrade you will need to set this up again.

PROC FSEDIT

PROC FSEDIT is one of the most powerful procedures contained in SAS/FSP® software. It allows for full-screen data entry, editing, viewing, searching, and querying. Data are displayed by observation.

FSEDIT may be invoked two ways. From the command bar, issue the command FSEDIT, along with the name of the data set you wish to edit. As with all SAS data sets, use a one-part data set name for a temporary data set, and two-part name, i.e., *libref.filename*, for permanent data sets. An example is shown below in Figure 3.

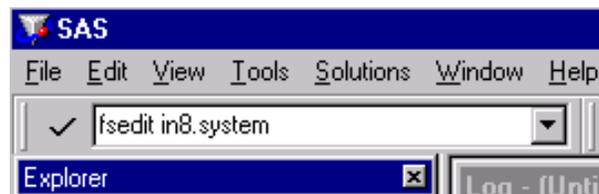


Fig. 3 - Invoking FSEDIT from the Command Bar

Alternatively, you may invoke FSEDIT from the Program Editor Window with a PROC FSEDIT statement that ends with a semicolon, as shown in Figure 4. The PROC statement requires one of two options: **DATA=filename** for editing existing SAS data sets, or **NEW=filename** for creating a new data set.¹ You may also add the **LIKE=** option, to create a data set that is identical in structure to an existing data set:

```
PROC FSEDIT NEW=newfilename
LIKE=oldfilename;
```

¹ This was originally in SAS/FSP but has been moved to Base SAS.

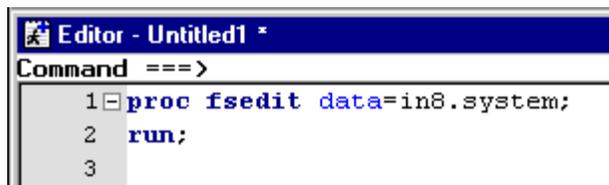


Fig. 4 - Invoking FSEDIT from the Program Editor Window

Figure 4 shows an example of the FSEDIT window. Note that the procedure and file name are indicated at the top of the window. The display may vary depending on the window manager being used.

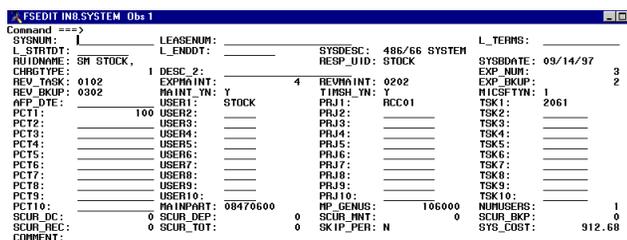


Fig. 5 - Example of an FSEDIT Window

Data Query

There are many ways to search for data within PROC FSEDIT. For example, if you know the observation number, from the View menu, select Observation number and enter the number in the dialog box that appears.

In many cases you will want to query the data based on information other than observation number. Using the Search menu, you can specify any of the following criteria for your query:

- a set of conditions (Where²... and Where Also...)
- the value of a field (Name... and Locate...)
- part of the value of a field³ (Search for Value...)
- the value of more than one field (Find...)
- embedded text strings⁴ (String... and Search For Value...)

For example, to look for a record with name Cyndie and case number “1”, select Find, which allows you to look for the value of more than one field.

To find an embedded string, e.g., “Cyn”, use the String and Search for Value commands.

FSEDIT also allows for WHERE processing. The WHERE clause can be specified at the time the procedure is invoked or you can select WHERE from the SEARCH menu to open up a WHERE dialog box shown in Figure 6. Enter the WHERE clause and click OK.

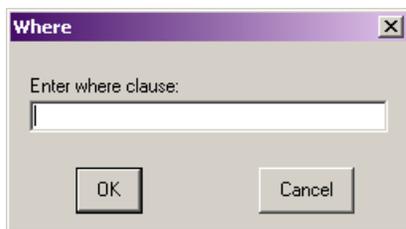


Fig. 6 - Specifying a WHERE Clause in FSEDIT

DATA EDITING

Editing in PROC FSEDIT is done in “real time.” There is no log of the changes you make, nor is there an “undo” function once you have edited a given observation. Make a backup of your data before you begin the editing process. To cancel changes within an observation, issue the CANCEL command before moving to the next observation or issuing the SAVE command.

The DELETE command may not physically remove observations from a file. To physically remove deleted records, create a new data set with a DATA step or with any SAS procedure that updates files in place, e.g., PROC SORT.

To add an observation either press the function key⁵ assigned to ADD, or from the Edit menu, select Add New Record. You may also add records by duplicating an existing record and making changes as needed. To duplicate a record, press the appropriate function key, or from the Edit menu, select Duplicate record. Note the following:

- Use the <Tab> key to move to and from the fields on the data entry screen.
- If you have no data to enter into a REQUIRED field, or attempt to enter invalid data into the data field, from the Edit menu, select Override to leave the observation.

CREATING A NEW SAS DATA SET WITH PROC FSEDIT

When you invoke PROC FSEDIT using the NEW= option, the procedure will appear in a data definition window as shown in Figure 7. As with an existing data set, the procedure and file name are indicated at the top of the window. Also note that the display may vary depending on the window manager being used.

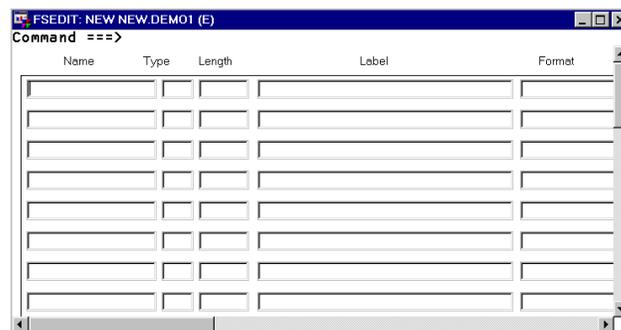


Fig. 7 - FSEDIT in Data Definition Mode

You create a data set by entering definition information as shown in Figure 8 below. Note the following rules for defining variables:

- Variable names must follow SAS naming conventions.
- Character variables must be assigned “type” c or \$.
- Each variable should be assigned a length.⁶ Eight is the default.
- You may assign a label, format, or informat. To view the Informat column, from the View menu, select Informat.

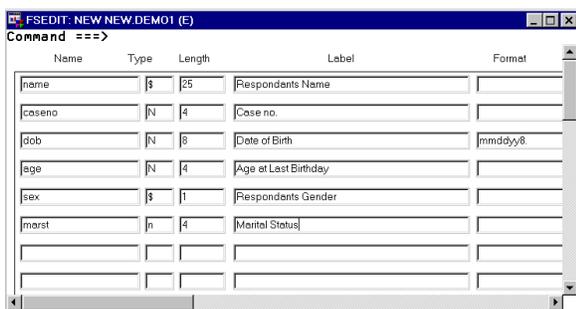


Fig. 8 - Data Definition Mode

After you are finished entering the information, type End on the Command line, or press the End function key to execute. A data entry screen will appear similar to the one below, with the variable names listed.

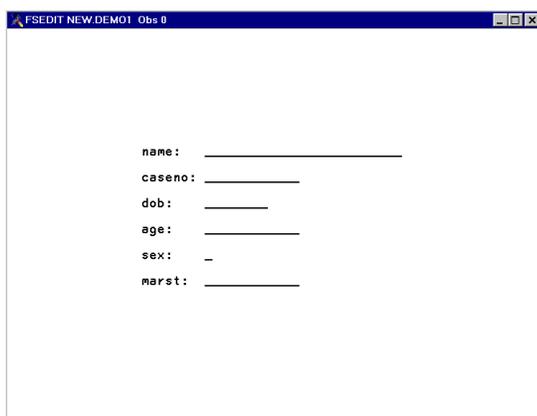


Fig. 9 - Default Screen

PROC FSVIEW

PROC FSVIEW is similar to FSEDIT, but it is a newer procedure, and more flexible. It is also part of Base SAS Software. Differences between FSVIEW and FSEDIT include the following:

- In FSVIEW data are displayed as rows and columns.
- FSVIEW can be used for both editing and browsing. The default mode is BROWSE.
- FSVIEW allows in-place data sorting.

To invoke PROC FSVIEW, type FSVIEW and the name of the data set at your command line, as shown in Figure 10.

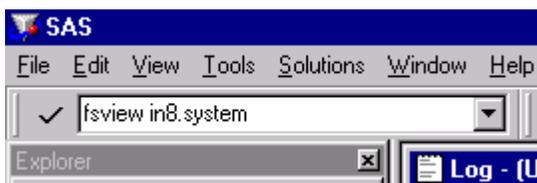


Fig. 10 - Invoking FSVIEW from the Command Line

Alternatively, you may submit the appropriate command from the Program Editor window. An example is shown in Figure 11 below. The PROC FSVIEW command, like the PROC FSEDIT command, requires that either a DATA=filename or a NEW=filename option be specified. Using the NEW= option will bring up the data entry window, just as with FSEDIT.

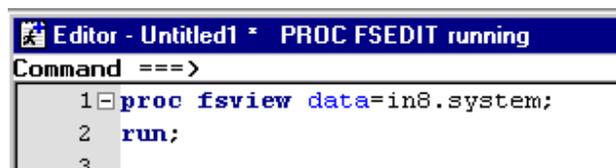


Fig. 11 - Invoking FSVIEW from the Program Editor Window

The FSVIEW window displays data as rows and columns. An example is shown in Figure 12 below.

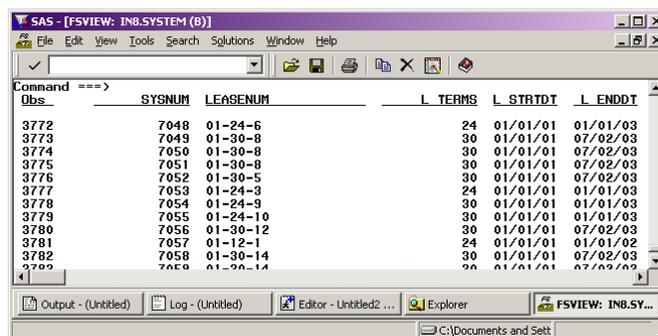


Fig. 12 - Example of an FSVIEW Window

Data Query

PROC FSVIEW is not quite as fully featured as FSEDIT for data query. You can search for a specific observation by number, or for a specific variable on the screen, from the Search menu.

FSVIEW also allows you to use WHERE processing to select observations for display. As with FSEDIT, you may select WHERE from the Search menu or type WHERE at the command line of the FSVIEW to open up the WHERE dialog box.

DATA EDITING

Like PROC FSEDIT, the FSVIEW procedure edits a SAS data set in place. Be sure to make a copy of the original data set before you begin editing.

In order to edit data with PROC FSVIEW you must first enter EDIT mode. You may do so by specifying the EDIT or MODIFY option in the PROC FSVIEW statement. If you did not do so, however, you can still switch into edit mode from within the FSVIEW window. Select Update from the Edit menu and then select RECORD (the default) or MEMBER level control. With record level control the observation being edited is locked, allowing you to open multiple FSVIEW windows for browsing or editing the same data set. In addition, other users can edit the same data set simultaneously (except for the locked observation(s)). With member level control the entire data table is locked. The table cannot be opened in another window or user. Note that with record level locking, an observation is not locked until it is selected.

Unlike FSEDIT, FSVIEW allows you to sort data in place. Issue the SORT command followed by the variable name(s) on the command line, or select Sort from the Edit menu. Data must be available for UPDATE in order to sort.

FSVIEW also allows you to define formulas to perform computations and manipulate data. Formulas can be specified to manipulate existing variables in the data set, or you can create new variables to display the results of the formula. Created

variables are not saved with the data—they are available only in the FSVIEW window. Issue the DEFINE command in the command line to bring up the DEFINE window:

For example, in the DEFINE window below (Figure 13), we have created a new variable, NEWRATE, which equals the variable SPECRATE*1.3. Note: you may also specify the formula at the same time you issue the DEFINE command.

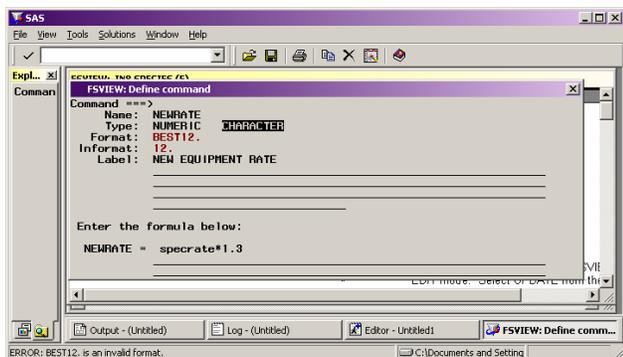


Fig. 13 - Example of an FAVIEW DEFINE Window

The formulas are stored in the SAS catalog entry with the type formula. Formulas may be reviewed by issuing the REVIEW command in the FSVIEW window.

CREATING A NEW SAS DATA SET WITH PROC FSVIEW

You can create a new data table with FSVIEW by using the statement

```
PROC FSVIEW NEW=filename
```

Or issue the NEW command from within the FSVIEW window. This code will invoke PROC FSVIEW in data definition mode as shown in Figure 8 above. As in PROC FSEDIT, you may use the **LIKE=** option to create a data set identical in structure to an existing data set.

VIEWTABLE

The Viewtable window provides an interactive way to create new tables, and to view or edit existing tables. This feature was introduced with Release 6.12 but required SAS/FSP to edit or sort tables. In Version 8 it has become part of Base SAS software. As with FSVIEW, data in a Viewtable window can be available for both editing and browsing. The default mode is BROWSE.

There are several ways to access the Viewtable window. In SAS Version 8, double-click the name of the data set you want to view in the Explorer window. You can also type VIEWTABLE (or "VT") and the name of the data set at the command line. By default the table is displayed as rows and columns, with variable labels as column names. To display the table with variable names, select Column Names from the View menu. Data may also be viewed in Form view by observation. To switch to Form view, select Form View from the View menu.

Data Query

Viewtable also allows you to use WHERE processing to select observations for display. Select WHERE from the Data menu or type WHERE in the command bar. This brings up a WHERE EXPRESSION dialog box, which allows you to build the

WHERE expression. The left-hand box displays variables that are available in the data set, or allows you to specify a constant. An example of this window is shown in Figure 14 below.

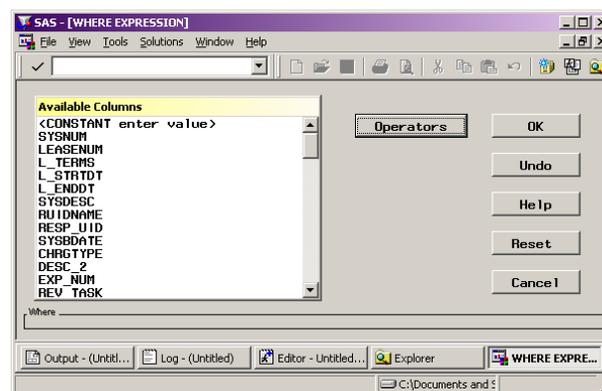


Fig. 14 - A Viewtable WHERE EXPRESSION Window

You can also define searches by selecting Find from the Edit menu. The dialog box that appears lists the variables in the data set, and allows you to specify Boolean operators (e.g., EQ, LT, GT) to create a specification to guide your search.

Data Edit

As mentioned, the default mode of Viewtable is browse only. To edit the data you will need to enter edit mode. From the Edit menu select Edit Mode. You are automatically given record-level control.

Within Viewtable you can add rows, copy rows, or delete rows by selecting the appropriate option from the Edit menu. Once you have defined the new row, select Commit New Row from the Edit menu.

To edit within an observation, type in the new value over the old value. One important advantage of Viewtable over FSEDIT or FSVIEW is the ability to undo edits. This is accomplished by selecting Cancel Row Edits from the Edit menu.

Like FSVIEW, Viewtable allows you to sort the data in place. Select the column you wish to sort by, right click, and select Sort. Select Ascending or Descending from the SORT dialog box.

You may also type SORT followed by the variable name at the command line or select Sort from the Data menu. Select the sort variables and other options such as output data set name in the dialog box. Using this method you can save the data to a new data set by selecting the Save As tab and entering a new table name. You may also select advanced sort options such as No Duplicates from the Advanced tab. An example is shown in Figure 15 below.

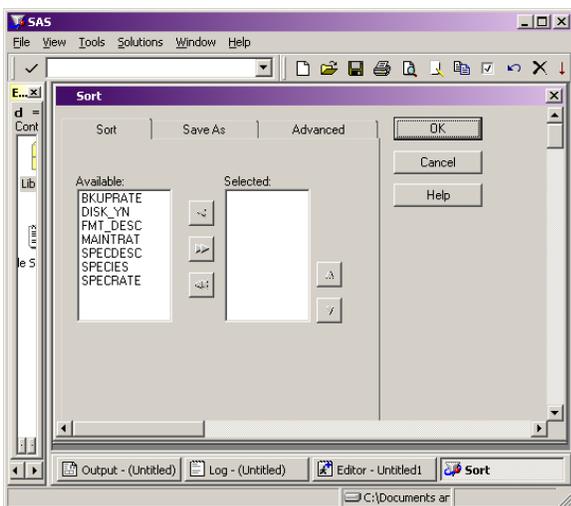


Fig. 15 - A Viewtable SORT Dialog Box

Creating a New Data Set with VIEWTABLE

If Viewtable is open, you can use the New icon on the toolbar or select New from the File menu to open a new data set. If Viewtable is not open, you can type VT or VIEWTABLE in the command bar. You can also right-click on the library in the Explorer window where you want to add the new data set and select New from the pop-up menu. A new blank Viewtable window appears, as shown in Figure 16.

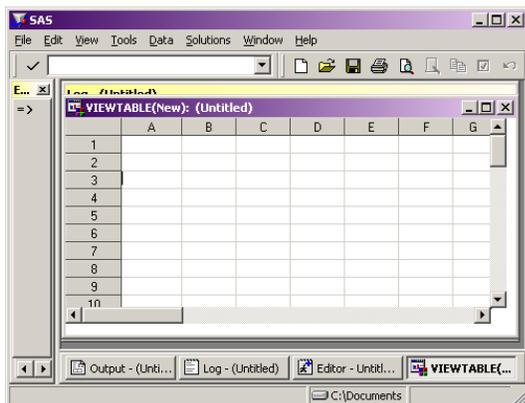


Fig. 16 - Creating a New Data set in Viewtable

To specify the attributes of an individual column, right click on the column and select Column Attributes. In the resulting dialog box, you may name the variable, select whether it is character or numeric, give it a variable label, and specify its length, format, and informat, as shown in Figure 17.



Fig. 17 - Specifying Column Attributes in Viewtable

For a list of valid formats and informats, click on the button to

the right of the appropriate entry field. You may then select the desired format or informat from a list.

Once you have set the variables' attributes, you can return to the main Viewtable window and enter your data interactively. When you close the data set, the observations you have entered will be saved.

During data entry, the data are only available for viewing in Table view. Once the data set is closed, the data may be viewed in either Form or Table view.

ANALYST APPLICATION

The Analyst Application⁷ is an interface to basic statistical analyses in the SAS System. While the Analyst application is generally considered a statistical application, it can be quite useful for interactive data manipulation. With this procedure you can view and edit data, perform data transformations, and create new variables. You can also save your data by overwriting your original data source, or create a new table by combining, summarizing, or transposing existing data tables.

To invoke the Analyst Application, from the Solutions menu, select Analysis, then select Analyst from the resulting submenu. The Analyst Application allows you to open any local file⁸, not just SAS Files. To open a file⁹, including non-SAS files such as an Excel spreadsheet, select Open from the File menu. Select the data type, Library and Table from a window such as the one shown in Figure 18, and click OK

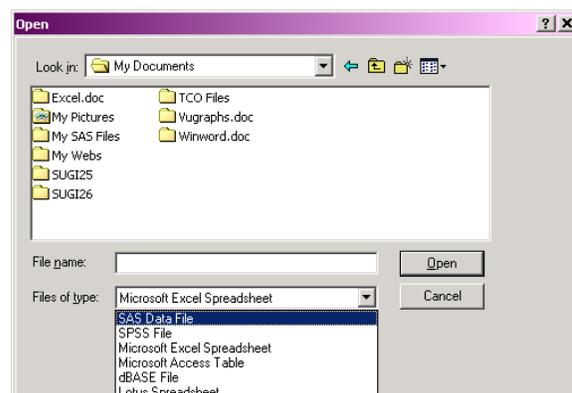


Fig. 18 - Selecting a Library and Table for the Analyst Application

The selected table comes up in browse mode, within a window that resembles a Viewtable window, as shown in Figure 19. Switch to edit mode by selecting Mode from the Edit menu, then selecting Edit from the Mode submenu.

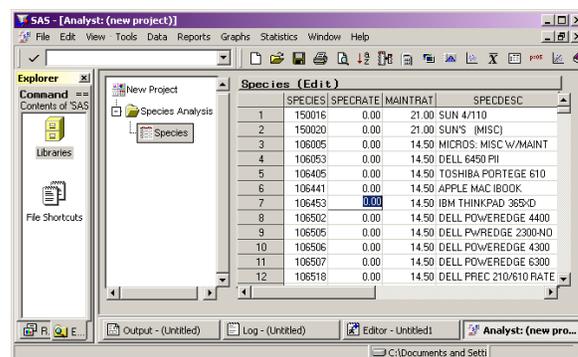


Fig. 19 - A Table in Edit Mode in the Analyst Application

Manipulating Data in the Analyst Application

To sort the data, select Sort from the Data menu. Then select the sort variable and type of sort (ascending or descending) and click OK. To subset the data, from the Data menu select Filter, then select Subset from the resulting submenu. A dialog box appears which allows you to select variables and specify Boolean operators to generate a WHERE clause to subset the data.

Data may also be recoded by selecting Transform from the Data menu, then Recode Values or Recode Ranges from the resulting submenu. The dialog box that appears will allow you to rename columns and to convert character variables to numeric or vice versa. Alternatively, you can convert numeric data to character or character to numeric by selecting Transform from the Data menu, then Convert Type from the resulting submenu.

As with FSVIEW, computed variables can be created in the Analyst Application. Select Transform from the Data menu, then select Compute from the submenu. From the dialog box that appears, select the new variable name, specify the formula, and click OK, as shown in Figure 20. The computed field is added to the table.

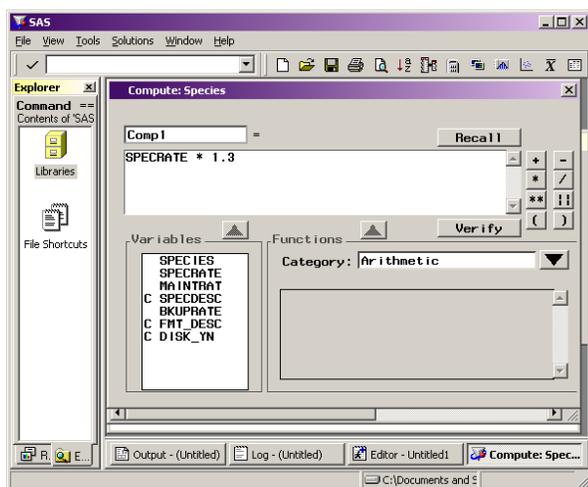


Fig. 20 - Computing a New Variable in the Analyst Application

Unlike FSVIEW, the new variable(s) can be permanently saved to the SAS Table. From the Edit menu, select Save. You can also save a record of the entire session, including the data table and any code generated by SAS to perform an analysis or create graphs and tables, as a Project. From the File menu, select Projects and enter a name for the Project. This feature allows you to edit in place while maintaining an audit trail of your work.

SUMMARY

PROC FSEDIT, PROC FSVIEW, Viewtable, and the Analyst application provide SAS Version 8 users with useful tools for easy data viewing and editing. Each of the four methods discussed in this paper has relative advantages and disadvantages when compared to each other. By selecting the one best suited to your particular needs, or by utilizing a combination of the four, you can greatly simplify the process of data viewing and editing with no corresponding loss of flexibility and power.

REFERENCES

SAS Institute Inc. (1999), *SAS ONLineDoc*®, Version 8, Cary, NC, USA: SAS Institute Inc.

SAS and SAS/FSP are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. Microsoft and Windows NT are registered trademarks of Microsoft Corporation in the USA and other countries. ® indicates USA registration.

Comments and questions may be sent to either of the authors:

Terry Fain
 RAND
 1700 Main Street
 Santa Monica, CA 90401
 Phone (310) 393-0411 x7804
 FAX (310) 451-7004
 fain@rand.org

Cyndie Gareleck
 RAND
 1700 Main Street
 Santa Monica, CA 90401
 Phone (310) 393-0411 x7815
 FAX (310) 451 7033
 cyndie@rand.org

¹ Custom screens may also be defined and used in FSEDIT. Because their primary usefulness is in data entry rather than data editing, we have not included a discussion of FSEDIT screens in this paper.

² Character data only.

³ You may also use the LOCATE: command from the command bar to search for partial values.

⁴ This is called a temporary WHERE clause. It can be canceled by clicking on Search, Where Undo.

⁵ To see or modify key definitions, from the Tools menu, select Options, then from the Options submenu, select Keys. If ADD is not listed, add it.

⁶ "Length" is the number of bytes of internal storage, not screen display length.

⁷ In addition to Base SAS, SAS/STAT® and SAS Graph must be installed on your system and SAS Assist must be licensed.

⁸ SAS/Access for PC File formats must be licensed and installed.

⁹ To select a current SAS table from the File menu, select Open by SAS Name