INTRODUCTION

The enhancements to Version 6 of SAS/AF® software add flexibility and power to computer-based training (CBT) applications. Enhancements to SAS/AF software for CBT courses include the powerful new Screen Control Language (SCL), extended questioning features, a multilevel, menu-linking facility, and the ability to vary screen size and color. New features that help in the development of a course include new printing options, additional course maintenance utilities, a user-friendly command menu, and the ability to test part of the course while in the interactive BUILD procedure. This paper describes and illustrates some of the new features that enhance both the development and the display of a CBT course.

CBT ENTRIES

Some of the new features available for CBT type entries are

- fill-in-the-blank questions
- the WRONG= option, which allows for branching
- multiple correct answers to a question
- the music interface
- the SAVE command.

With Version 6 enhancements, you can easily build fill-in-the-blank questions in addition to multiple-choice questions. By using the CORRECT= option on the frame indicator line, you can specify a correct answer of up to thirty-two characters, with or without embedded blanks. The example below illustrates the syntax of the CORRECT= option. If the answer contains embedded blanks, you must enclose it within single quotes.

```
Correct= zoom
IIhat display manager command enlarges the active window?

Very good.
The ZOOM command enlarges the active window to fill the
screen. It also reduces an enlarged window to it's default
size.

No, think of a camera.
```

Screen 1 A Fill-in-the-Blank Question

The WRONG= option is also new to CBT entries. Use the WRONG= option to specify where you want students to be branched if they use up all allotted tries for a particular question. For example, if you allow students two attempts to answer a question, you can provide a hint following the first incorrect answer, then branch them to remedial instruction following the second incorrect answer.

```
Wrong=cbt.fun.no1.cbt
```

You can design a multiple-choice question with multiple correct answers. If a student chooses at least one correct answer, he receives the message "Your answer is partially correct".

Another new CBT feature is the MUSIC= option. A tune is played either when a frame first displays or as a result of a student's answer choice. Specify the tune by listing each note and its duration in the MUSIC= option.

The SAVE command allows students to exit a course and the SAS System in the middle of a lesson and automatically return to the place they left off when they re-invoke the course.

MENU ENTRIES

With the multilevel, menu-link facility, several menus can be linked, thus increasing the number of selections available to students from a primary menu. A student experienced with the structure of a course can make a selection from a submenu while on the primary menu even though the selection is not listed on the primary menu.

For example, if a course has a main menu and three lesson menus, you can link the menus together so that a student can select item 2 from the Lesson 1 menu by typing 1.2 on the main menu.

To link items on a menu, you use a two-step process. First, you specify which menus to link by placing an X in the Menu-Link field on the attribute panel of the higher-level menu.

```
MLINK: ADD.MENU (F)
```

```
Option  Name  Type  Library  Catalog  Menu-Link
1  LESSON1  MENU  *  *  X
2  LESSON2  MENU  *  *  X
3  LESSON3  MENU  *  *  X
```

Then, to establish the menu link, issue the MLINK command on the catalog directory where you name the highest level menu of the group you want to link.
You can specify the number of levels of menus you want to link using the LEVELS= option. The default is one level. You can also establish the menu link using the MLINK statement with the PROC BUILD statement.

PROGRAM ENTRIES

New PROGRAM entry features that support computer-based training applications include the following:

- multipage displays
- additional parameters on the attribute panel
- Screen Control Language.

Multipage Displays

When you are designing a PROGRAM entry, the amount of text being placed on the display panel may not fit on one screen or page. Define multiple pages that are presented one at a time when the screen is displayed. The dashed line is used to delimit each page. When the screen is displayed, a new page is presented each time the screen is scrolled up or down.

Additional Parameters on the Attribute Panel

The attribute panel for a PROGRAM entry supports how fields are displayed, validated, and processed. Additional attribute parameters include

- PAGE (the page of the display panel on which the field appears)
- ALIAS (an alias name used to reference the field in the program)
- CURSOR (positions the cursor on the field when the screen is displayed)
- ERROR COLOR (highlight color for the field if in error)
- ATTR (extended highlighting definition for the field if in error)
- INITIAL (initial value for the field when first displayed)
- REPLACE (text to be inserted in the program if the field is filled in)
- AUTOSKIP (positions the cursor on next field when the current field is filled in).

Screen Control Language

Screen Control Language (SCL) combines student input with information obtained from statements within the SCL program to control screen interaction and other aspects of the computer-based training application.

Capabilities Every aspect of the CBT application, from preparing the training environment, pretesting, choosing instructional paths, monitoring student interaction, validating student responses, tracking student performance, posttesting, to terminating interaction with the SAS System is controlled by SCL. Specifically SCL is used to

- control the computer-based training application environment
- access SAS files and obtain information from them to support student tracking
- control communication between the screens in the CBT application
- control interaction between the student and the CBT application
- perform field validation and cross-field validation of student responses
- provide error handling and customization of messages
- submit SAS® programs for execution
- interact with the SAS System.

Language Components SCL includes statements, functions, and system variables. SCL statements control when and how other SCL statements and functions execute, provide field validation and error checking capabilities, and submit SAS programs for execution. SCL functions provide specialized actions or tasks like monitoring the field entry and checking if it is in error. SCL variables provide communication between the SCL and the display such as writing messages.
SCL is very DATA-step like and includes most of the statements and functions that make up the DATA step. In addition, more than one hundred additional statements, functions, and special system variables make up the language.

Example: Check the symbolic field \texttt{WORD} for the value \texttt{CHART}. If the value is incorrect, position the cursor on \texttt{WORD} and write a message.

\begin{verbatim}
IF WORD ^= 'CHART' then do;
   cursor WORD;
   msg_='Your answer is incorrect. Please try again.
end;
\end{verbatim}

The statements, functions, and system variables available in SCL support:

- diagnosing and debugging errors
- controlling fields (that is, cursor positioning, field error, protection, and so forth)
- working with SAS data sets (that is, opening, closing, sorting, and so forth)
- displaying special windows (selection lists) and choosing entries from those windows
- working with SAS variables and observations (that is, locating, reading, writing, and so forth)
- processing command strings (that is, executing commands and returning nth word)
- working with external files (that is, allocating, verifying, and so forth)
- integrating macros
- processing character strings
- using structured programming logic
- interacting with and controlling windows.

Methodology

The SCL program is prepared on the source panel for the PROGRAM entry it supports, issue the SOURCE command on the command line of the Display panel to display the Source panel.

\texttt{Command --- SOURCE}

After the SCL program is written, issue the COMPIL command. When the program is compiled, intermediate executable code is stored.

Screen 5 Compiling the Program

A MESSAGE window contains any compile warnings and errors.

The program does not execute until the PROGRAM entry is presented in display mode.

SCL Structure

Unlike traditional SAS programs, SCL programs require determination of when statements in the program are to be executed. Program execution is controlled by grouping statements using reserved SCL labels. These labels indicate when to execute the statements in that label.

Every application has three phases:

- a start-up or initialization phase
- an execution or main processing phase
- an ending or termination phase

Similarly, SCL programs have an initialization, execution, and termination phase.

\textbf{INIT:} The initialization phase sets default values, prepares and displays messages, initializes variables, and opens files. The tasks that are placed in the initialization phase are executed before the PROGRAM entry is displayed.

\textbf{MAIN:} In the main phase the student interacts directly with the PROGRAM entry. The student is prompted for information, responses validated, fields checked, and messages written, and other programs call that prompt for additional information.

\textbf{TERM:} In the termination phase the application ends. Housekeeping or clean-up tasks are performed like resetting default values and closing files. The tasks in this phase are executed when the PROGRAM entry is terminated.
Illustrations

Example: The SCL program prompts the student for a response. If it is correct, the student is transferred to a CBT entry. Otherwise, the student is prompted to correct the entry.

Program entry display panel:

Screen 6 Display Panel

SCL program:

Screen 7 Source Panel

Screen 8 Source Panel (Continuation)

Example: The SCL program prompts the student for a response. If it is correct, the student is prompted to continue. Otherwise, the student is prompted to correct the entry. A hint is provided.

Program entry display panel:

Screen 9 Display Panel
AUTHORING FEATURES

Some other features available to you when developing CBT courses include the ability to:

- set sizes and colors of windows
- enhance printing
- test the application
- use command menus.

Setting Sizes and Colors of Windows

Among the enhancements available to all entry types is the ability to set the size of any window. In a multiple-window display, you can allow students to see related windows at the same time. For example, you may want to design a HELP window that is displayed on a corner of the screen. The size of the window is set by entering the coordinates on the general attribute panel or by using the WGROW and WSHRINK commands with the SETWSZ command.

In addition to size, you can control the color of an entry's background, border, command area, and command line. A new entry type, EDPARMS, allows you to set colors and other attributes for all new CBT, MENU, HELP or PROGRAM entries.

Printing

The PRINT statement used with the BUILD procedure allows you to print any or all parts of an entry. The DISPLAY, SOURCE, and ATTR options print the display, source, and attribute panels. The LISTDIR option prints the catalog directory; the XREF option prints cross-reference tables showing the entries called by a menu. The SHOWPAD option prints PROGRAM entry display panels as they appear to the user. The ET= option specifies an entry type for printing. In addition to selecting the entries you want to print using the SELECT= option, you can now use the EXCLUDE= option to indicate the entries to exclude.

Testing the Application

With Version 6 of SAS/AF software, you can display parts of your application as you build it. To display an entry while in the BUILD procedure, simply type TESTAF on the command line of the current entry. For example, to test a PROGRAM entry, issue the TESTAF command after the program has compiled. When you finish testing the entry, issue the END command, and you are returned to the PROGRAM entry in edit mode.

Using Command Menus

For new or infrequent builders of CBT courses using SAS/AF software, command menus can be helpful. A command menu, which can be displayed in place of the command line, lists all the current commands available for that window.

The active command is indicated in reverse video. To issue a command, move to that command in the list using the TAB or FIELD key; then press ENTER. In addition to the list of commands, a description of the active command is displayed in the line below the command menu.

The example below illustrates the command menu for the catalog directory. The COPY command is the active command. If you press ENTER to issue the COPY command, a small window appears to prompt you for the information of the source entry and the target entry.
## CONCLUSION

The enhancements to Version 6 SAS/AF software increase the power and versatility of a CBT course, as well as making the CBT course easier to develop, document, and maintain.

### References


SAS and SAS/AF are registered trademarks of SAS Institute Inc., Cary, NC.