HyperCard: Debugging Scripts (11/92)

Article Created: 16 November 1990
Article Reviewed/Updated: 30 November 1992

TOPIC ----------

HyperCard provides some useful tools for troubleshooting your scripts. These debugging tools let you step through a handler line by line as it runs, inspect the values of variables, and trace the flow of messages.

DISCUSSION ----------

Debugging a Handler

To debug a handler, follow these steps:

1) Open the script that contains the message or function handler you want to debug.

2) Click to place the insertion point on the line with the on or function statement that defines the handler.

   Or, depending on where you want to start, place the insertion point on any statement of the handler that is not a comment.

3) Choose Set Checkpoint from the Script menu (or press Command-D) to set a temporary checkpoint.

   A checkpoint tells HyperTalk where you want to start watching the script as it runs.

4) Press Enter to save the script and close the script editor window.

5) Perform the action in HyperCard that activates the script (for example, click a button).

   When HyperCard hits the checkpoint, it opens the script and puts a box around the current statement. HyperCard also displays the Debugger menu.

   HyperCard does not enter the script editor itself: All HyperCard menus remain in the menu bar. (Since the script is still running, HyperCard must preserve the current context.) But only the Debugger menu is active.

6) Choose commands from the Debugger menu to proceed.

The Step Menu

- Runs the current statement and moves to the next one.

- If the current statement calls another message or function handler, Step does not switch to the new handler and step through it. It just gets the result of that call.

Step Into

- Runs the current statement and moves to the next one.

- If the current statement calls another message or function handler, Step Into does switch to the new handler so you can step through it.

Trace

- Runs through the rest of the statements one line at a time, but without pausing after each statement.

- If the current statement calls another message or function handler, Trace does not switch to the new handler and trace it.

Trace Into

- Runs through the rest of the statements one line at a time, but without pausing after each statement.

- If any statement calls another message or function handler, Trace Into will switch to the new handlers and trace them as well.

Go

- Runs the remainder of the script from the current statement without stepping.

- After a Go command, HyperCard leaves the Script Editor window open, but activates the window containing the current card.

Trace Delay...

- Displays a dialog in which you can specify the number of ticks that HyperCard will pause between each statement as it traces a handler.

Set Checkpoint

- Places a checkpoint at the current line of a handler.

- This command changes to Clear Checkpoint if you've already set a checkpoint on the line.

Abort

• Stops the script from running. After an Abort command, HyperCard leaves the Script Editor window open and active.

Variable Watcher

• Shows or hides the Variable Watcher, a window that allows you to inspect the values of parameter variables or global and local variables as they are set during the execution of a handler.

• To change the value of a variable using the Variable Watcher, click to select a variable. Its value appears in the bottom panel. Edit the value and press Enter to save it.

Message Watcher

• Shows or hides the Message Watcher, a window that allows you to watch what messages HyperCard sends as it runs the script.

• For example, if you step through a statement such as doMenu "New Card", the Message Watcher will show the closeCard and openCard messages that run when the new card is added to the stack.

Keywords: <None>

==========

This information is from the Apple Technical Information Library.

19960215 11:05:19.00
Tech Info Library Article Number: 14074