Tech Info Library

HyperCard, Apple Events, and Frontier (8/92)

Revised: 3/25/94
Security: Everyone

HyperCard, Apple Events, and Frontier (8/92)

==========

Article Created: 4 June 1992
Article Reviewed/Updated: 25 March 1994

TOPIC --------------------

When using the "send" command to execute Frontier scripts while the scripts are running, status messages are sent to the Frontier status bar using the msg command in Frontier. Is it possible to have the messages sent to the Frontier status bar sent to a HyperCard field (or "status window") instead?

DISCUSSION --------------------

HyperCard is not multi-threaded. In particular, it can't automatically handle Apple Events while it's waiting for a reply to an Apple Event that it sent itself. While the "send" command awaits a reply from Frontier, all HyperCard can do is to handle update events from the Event Manager, suspend and resume events from the Process Manager, and command-periods from the user.

However, you can use an XCMD to do what you want in one of two ways:

- Send the Apple event to Frontier in queueReply mode and handle the reply that comes back from Frontier as a separate Apple event in an Apple event script (class 'aevt', id 'ansr'). This would mean that HyperCard can respond to events coming from Frontier in the meantime, including events to display the Frontier status messages in HyperCard fields. It would also mean that HyperCard will be able to respond to user events while it awaits the reply from Frontier, so you'll have to do some careful HyperTalk scripting to figure out how to reconcile what the user is doing with what you want to do when the reply from Frontier comes back. The SendAppleEvent XCMD, available online, will send an event in queueReply mode.

- Write an XCMD that sends the Apple event to Frontier and calls GetSpecificHighLevelEvent in the idle proc for AESend, in order to extract 'do script' events for HyperCard to handle while it awaits the reply from Frontier. Writing this XCMD would require pretty good

knowledge of the High Level Event Manager, the Apple event Manager, and the 2.0 XCMD callbacks, particularly RunXHandler.

This article is adapted from the Claris Tech Info database.
Support Information Services

Tech Info Library Article Number:14511