

Why do we need IsDialogMessage at all?

 devblogs.microsoft.com/oldnewthing/20120416-00

April 16, 2012



Raymond Chen

alv wonders why we need the IsDialogMessage function at all. “All its activity could take place inside the window procedure of the modeless dialog itself”, since when it doesn’t have focus, it shouldn’t be responding to messages anyway. Sure, that works great if the modeless dialog has focus. But it almost never does. What has focus is a control *inside* the modeless dialog. And in that case, the modeless dialog never sees the message, since the rule is that keyboard messages go to the window with focus. And that ain’t the modeless dialog box. Consider, for example, a message box with OK and Cancel buttons. Focus defaults to the OK button. You hit the TAB key to move to the Cancel button. The TAB message goes to the OK button because it is the control with keyboard focus. The OK button says, “I don’t know what TAB means. I guess I’ll just beep.” (The button control doesn’t know whether it’s part of a dialog box or not; it just worries about being a button.) The `IsDialogMessage` function is in the message loop so that it can intercept the TAB message *before it reaches the OK button*. At this point, `IsDialogMessage` can step in and say “Hang on, I’ll take care of this” and use the TAB key to perform dialog box navigation. (Of course, as we know, it first checks with the control that stealing the message is okay.)

Some programming frameworks capture the concept of “stealing messages” with names like `PreTranslateMessage` and event routing. But the basic idea is the same: Letting somebody other than the target of a window message get a chance to intercept the message and do something special with it.

Raymond Chen

Follow

