

Using the TAB key to navigate in non-dialogs, redux

 devblogs.microsoft.com/oldnewthing/20131009-00

October 9, 2013



Raymond Chen

You want to use the TAB key to navigate through a non-dialog, so you call `IsDialogMessage` in your message loop, but it doesn't work!

```
// code in italics is wrong
MSG msg;
while (GetMessage(&msg, NULL, 0, 0)) {
    if (!msg.hwnd || !IsDialogMessage(msg.hwnd, &msg)) {
        TranslateMessage(&msg);
        DispatchMessage(&msg);
    }
}
```

The problem here is that you are passing the wrong window handle to `IsDialogMessage`. The first parameter to `IsDialogMessage` is the dialog-like window you want to be able to navigate through. But the code above passes the window that received the message, so you are basically telling the control to do TAB navigation *within itself*. And naturally, the result of that is that focus *stays where it is*, because if you ask a button, “Hey, could you move to your next tab stop?” the button is going to say, “Dude, I’m the only tab stop on my window!”

If you think about it, passing the window the message was already going to be dispatched to isn't very interesting. If that was the design of the function, then all the work of `IsDialogMessage` could've just been put in `DefWindowProc` and we wouldn't have needed a `IsDialogMessage` function in the first place.

And if you think about it, passing the window the message was already going to be dispatched to isn't very interesting. If that was the design of the function, then the parameter isn't necessary. The function could just infer it directly from the `MSG` structure you passed as the second parameter.

The fix is to pass the main window handle to `IsDialogMessage` :

```
MSG msg;
while (GetMessage(&msg, NULL, 0, 0)) {
    if (!msg.hwnd || !IsDialogMessage(hwndFrame, &msg)) {
        TranslateMessage(&msg);
        DispatchMessage(&msg);
    }
}
```

Reminder: As noted in the original article, you should not use the WM_USER or WM_USER + 1 messages in your custom window.

Raymond Chen

Follow

