

# Re: Message Reflection

---

*Source:*

<http://www.tech-archive.net/Archive/Development/microsoft.public.win32.programmer.ui/2005-06/msg00164.html>

---

- *From:* "John Carson" <jcarson\_n\_o\_sp\_am\_@xxxxxxxxxxxxxxxxxx>
  - *Date:* Tue, 14 Jun 2005 11:52:05 +1000
- 

"Steve Thresher" <steve.thresher@xxxxxxxxxxxxxxxxxx> wrote in message  
[news:ePXSC5EcFHA.3464@xxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:news:ePXSC5EcFHA.3464@xxxxxxxxxxxxxxxxxxxxxxxxxx)

"John Carson" <jcarson\_n\_o\_sp\_am\_@xxxxxxxxxxxxxxxxxx> wrote in message  
[news:u0OBGHDcFHA.580@xxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:news:u0OBGHDcFHA.580@xxxxxxxxxxxxxxxxxxxxxxxxxx)

"Steve Thresher" <steve.thresher@xxxxxxxxxxxxxxxxxx> wrote in message  
[news:eLuvi5CcFHA.3808@xxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:news:eLuvi5CcFHA.3808@xxxxxxxxxxxxxxxxxxxxxxxxxx)

Can anyone tell me how message reflection should be implemented or point me at a reference on the topic. I am NOT using MFC. I am attempting to custom draw a header control and need to know how to get the WM\_NOTIFY message from the parent control to my window procedure for the header control. I can subclass the parent of the header control when it is created but what happens when you destroy the header control? The parent window is still sub-classed with no way to work out the address for the parents original window procedure. Any help would be greatly appreciated.

Steve.

I am puzzled by your question. When you subclass a window procedure, the address of the original window procedure is received as the return value of SetWindowLong. You need this address so you can call the original window procedure for those messages you don't choose to process. You can choose to remove the subclass when you destroy the header control for the sake of a slight efficiency gain, but it should continue to work if you just leave it there unless you have written it in some strange way. --  
John Carson

You MUST NOT unsubclass a window on operating systems prior to XP / 2003 as you cannot guarantee that no-one else has subclassed the window; XP / 2003 provide functions to subclass / unsubclass a window.

You could presumably check (using GetWindowLong) if the current window procedure is the one you set. If it is, then no-one else has subclassed it after you. If it isn't, you leave the subclass in place. Easier still,

## Re: Message Reflection

just leave it in place.

Also once the header control has been destroyed, so has the data attached to the window which includes the original window procedure address.

I am still puzzled.

MFC includes some kind of message relection technique which I was hoping to implement in pure Win32 code. I was just hoping someone would have some idea how to do it.

When it receives the WM\_NOTIFY method, the parent checks the ID of the sender to see if it comes from the header control. If so, it calls

```
return SendMessage(((NMHDR)lParam).hwndFrom, message, wParam, lParam);
```

If you want reflection to every child control, you can skip the check.

--

John Carson

.