
Subject: Re: OOP Programming, Was: Something Else
Posted by [steinhh](#) on Fri, 06 Mar 1998 08:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Just one thing about this kind of solution (from Mark Rivers):

```
> pro mca_display_event, event ; event processing for MCA application
>
>   widget_control, event.top, get_uvalue = mca_display
>   mca_display->event, event
> end
```

I realize(d) (after I started to write this message :-)) that this is the event processing for the entire application.... But for compound widgets, etc, it is customary (and a good custom, too!) to leave the compound widget ID's UVALUE to the "owner's" discretion. Thus, in general, it is wise to store the object pointer on the uvalue of the *first child* of the event.id.

I.e., in a compound widget event *function* (allowing the event to be modified and sent further up the hierarchy, or being gobbled up by sending along a zero value) one should do something like (untested code follows!):

```
function cw_object_event,event
    widget_control,widget_info(event.id,/child),get_uvalue=objec t
    return, object->event_func(event)
end
```

The observant reader will of course have noticed that if this scheme is followed *consistently*, you only need *one* event function (outside the individual class scopes, that is!) for *all* of your "objectified" compound widget types!!

Neat, if I may say so myself.

Darn, why can't I have some time to spend actually *doing* this OO stuff, instead of just writing about it!

Regards,

Stein Vidar
