
Subject: Re: Passing info and destroying widgets...
Posted by [rmlongfield](#) on Tue, 22 Jun 1999 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

In article <7kn0h5\$s0c\$1@news.doit.wisc.edu>,
"Liam Gumley" <Liam.Gumley@ssec.wisc.edu> wrote:
> David Fanning <davidf@dfanning.com> wrote ...
>> Liam Gumley (Liam.Gumley@ssec.wisc.edu) gives us an
>> example of a program that can record the last instance
>> of a button push in a non-blocking, non-modal widget ...(stuff cut)
but
>> by doing so you violate every tenet of good object programming
>> practice, in which the data should be encapsulated and
>> unseen by the outside world. Sucking the pointer out of
>> a widget program, except perhaps in the hands of just the
>> best programmers, is a practice that is guaranteed, it
>> seems to me, to get most of the rest of us in a hell of

This is really funny. As I was reading Liam's suggestion, I was thinking that this was a fantastic idea and something that I would certainly try to implement. Then, I go on to read David Fanning's comments and am dismayed. Well, you can be sure that I will think think carefully if I ever decide to use it.

I've recently been forced to deal with too many dangling pointers because my image tool gave me errors after I was creating (and not destroying) too many of them. I also now have two delete event handlers for each top level widget I create. One is called with a quit button and explicitly deletes all the pointers I have created. The second one is a cleanup routine (learned from DWF's book) which destroys the toplevel ID:

```
PRO Image_panel_cleanup_event,lasteventID
help,lasteventID,/structure
print,'At the very end of Image cleanup'
```

```
WIDGET_CONTROL,lasteventID,GET_UVALUE=image_infoPtr
PTR_FREE,image_infoPtr
END
```

This removed the errors related to too many pointers. Using HEAP_GC just got me into trouble because I didn't really know what I was deleting or why. I feel happier that my explicit deletion works well and I know why.

Actually, using Liam's pointer solution might enable me to avoid using the COMMON statements I just put into my programs to make my data accessible to other programs. It's a difficult choice....

Rose

Sent via Deja.com <http://www.deja.com/>
Share what you know. Learn what you don't.
