
Subject: Re: Widgets: group leader and procedures
Posted by [David Fanning](#) on Wed, 07 Jan 2004 04:20:49 GMT
[View Forum Message](#) <> [Reply to Message](#)

Jeff Guerber writes:

> I disagree: If you use a pointer for the widget state, you can change
> its contents and not have to worry about stuffing it back into the uvalue
> before you exit the event handler. Plus you won't be constantly making
> copies of everything in your state structure, which can be an advantage if
> it's large. Just remember to have the TLB's cleanup routine free pstate
> (I call it statep, myself). After I started writing widgets, I very
> quickly switched over to state pointers (until I discovered object
> widgets, that is!).

If you are going to the trouble of putting it back
in the UValue of the TLB, then a simple NO_COPY
will prevent any copying of the state structure
as you move it into and out of an event handler.

I personally prefer to write the NO_COPYs for
simple programs rather than deal with the cumbersome
pointer syntax. But there are distinct advantages to
a pointer in more complicated programs where the state
structure sometimes has to be passed around to programs
outside the confines of the normal single widget program.

But if the pointer syntax doesn't push a user quickly to
object widgets, then the user has a LOT more patience than
I have. :-)

Cheers,

David

--

David W. Fanning, Ph.D.
Fanning Software Consulting, Inc.
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
Toll-Free IDL Book Orders: 1-888-461-0155
