

---

Subject: Re: Communication between top-level bases.  
Posted by [mirko\\_vukovic](#) on Tue, 02 Jun 1998 07:00:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

In article <MPG.fdce647de8322e89897c4@news.frii.com>,  
davidf@dfanning.com (David Fanning) wrote:

>  
> David Foster (foster@bial1.ucsd.edu) writes a lot of excellent  
> advice about widget programming, including this:  
>  
>> A much more elegant way to share data \*within\* a widget is to create  
>> a STATE structure (I always call it STATE{} to avoid confusion) which  
>> contains the data you want to share. Then, before you register it  
>> with the XMANAGER, copy this structure into the UVALUE of the  
>> top-level widget:  
>>  
>> widget\_control, base, set\_uvalue=state, /no\_copy  
>  
> I couldn't have written this part of the article any better  
> myself, and (as Dave mentions) I suggest something very similar  
> in my IDL Programming Techniques book. But I developed those  
> ideas nearly two years ago now, and I find myself using  
> something quite a bit different today. Poor Dick, who knows  
> I am working on ideas for the next programming book, calls  
> me about every other day to see what our Current Best  
> Practice is now. When I am in the writing mode, it  
> changes frequently. :-)  
>  
> But what I do today is put the state structure (I often call  
> it the "info" structure) in an IDL pointer variable, like this:  
>  
> statePtr = Ptr\_New( {image:image, drawID:drawID, ...}, /No\_Copy )  
>

:-)  
I couldn't have written Dave's contribution any better (now does that sound familiar or what), as I dumped the State into Heap, oh say last Friday. As Dave pointed out, using /no\_copy is an invitation for programming lapses. :-)

I played for a minute or two (most of those due to editing and un-doing the edits, the other couple of seconds on thinking), about putting State (which I call something else) into an \*object\*. A jump from a structure to an object is a small one.

This again would involve cleanup upon exit, but one benefit is a cleaner pointer de-referencing syntax in the code. It would be able to provide for some

