
Subject: Re: Unable to get the user value of the top level base after a draw widget expose event

Posted by [MKatz843](#) on Fri, 15 Aug 2003 03:03:11 GMT

[View Forum Message](#) <> [Reply to Message](#)

I had this same problem for a while. You're event handler "checks-out" the state variable (like a library book), and while it's checked-out, some other event comes along looking for it.

If you take Pavel's advice and drop the /no_copy, you might run into a problem in which two different routines are setting values of the state and they may conflict or lose information somehow. You know that there is some kind of conflict going on where two routines are already reaching for the same state value.

The very best solution to all of this, a la Prof. Coyote, is to use a pointer for the state. You don't need to change much, just instead of state, the first time you set the uvalue use ptr_new(state, /no_copy). Then every time you get the uvalue, you'll be getting a pointer to a structure that only lives in memory once.

Your code becomes:

```
widget_control, event.top, get_uvalue = state
(*state).ThresOf = event.value
```

The line that re-sets the uvalue with state when you're done is no longer necessary, since you never "checked it out."

In this way, multiple events _can_ access and interact with the state with less fear of conflict.

One extra bit of advice. If *state is large, or you use many widgets, you should make sure to clean up properly when the widget is killed. Set a widget cleanup routine to free the pointer (if it's valid) before destroying the widget and losing the pointer reference. You'll avoid a potential memory leak.

I hope this helps.

M. Katz

Pavel Romashkin <pavel_romashkin@hotmail.com> wrote in message news:<3F3BAB5C.77984022@hotmail.com>...

- > It seems you are losing your State structure. To check, get rid of
- > /no_copy everywhere and try again. If it works, comb through your code
- > and find where do you forget to put State back in. Typically this
- > happens when you put your widget_control, event.top, set_uvalue = state,

> /no_copy in an IF statement that gets skipped in the code.
> Also, check if you don't forget to retrieve it in the pro in question.
> It should work just fine.
>
> Good luck,
> Pavel
>
