
Subject: Re: WIDGETS

Posted by [peter](#) on Thu, 01 Sep 1994 21:03:57 GMT

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

Joseph M Zawodny (zawodny@arbd0.larc.nasa.gov) wrote:

: No, no, no! You should avoid using common blocks in widget programming. I
: sense and desense widgets of all sorts routinely. My personal preference
: is to create a structure with one of the fields set to be a long integer
: array and I store the widget id's of the widgets I need to manipulate or
: (de)sensitize in this array (the structure is required only if you need to
: pass other data besides the widget id's). That way a simple

: widget_control,event.id,get_uvalue=array

But how do you sensitize/desensitize a widget that you didn't get an
event from? Suppose I perform an action that requires, say, the fields
in another base widget to change. I have to have their widget IDs, so
I need a common block, unless I store all of them in the uvalue of
every widget that might require them to change (which seems a little hard to
maintain (ah, the quick and dirty programming thread again!)).

: statement gets the info I need without using common blocks. The problem is
: that you might have multiple versions of the same widget running
: concurrently. If you do and use common blocks to pass data between
: widgets, the common blocks can (and do) get scrambled.

This is a problem -- I've only ever written applications where unique
instances of widgets are allowed, in which case a common block to hold
widget IDs, and another one to hold the panel state (i.e. the values of
variables that affect that panel) seems to work very well indeed.

Peter
