Subject: Re: auto updating of a nonmodal dialog widget
Posted by davidf on Tue, 08 Feb 2000 08:00:00 GMT
View Forum Message <> Reply to Message

Brian Reardon (reardonb@my-deja.com) writes:

> I have a non modal dialog widget composed of cw_field's and a few
> buttons. The cw_fields are automatically filled with data from an info
> structure when the dialog widget is created. The idea behind the widget
> is to allow the user to input the desired information into the fields,
> hit the APPLY button and have the
> information in the fields sent to the appropriate info structure.
> Everything is working fine. However, I also have a button that allows
> the user to select a file to read in the same information to the same
> info stucture. That button is also working fine. My problem is this,
> when the user reads in a file how do I automatically update the
> information in the cw_field widgets so that the user can see the data
> was read in correctly? Do I have to destroy the dialog widget and then
> create it again?

The technique I like to use to communicate between non-modal
widget programs is to use -- drum roll -- widget events!
I find this a natural idea in widget programming. :-)

It works like this. Suppose I have a program that is suppose
to do something, like--maybe--load color tables. And I have
another program that would like to know about that particular
something. (Maybe that program would like to redisplay an
image so that it appears in the colors of the color table
just loaded.) I have some mechanism (usually a keyword) to
tell the first program the identity of a widget that should
be notified when the "something" occurs.

With my XCOLORS program the code looks like this:

```
XCOLORS, NotifyID=[widgetToNotify, topLevelBase]
```

I store this information in XCOLORS. Then, when I finally
get around to loading the color tables I look to see if
there is somebody I am suppose to notify. If so, I create
a widget event and send it to them. In this case, my event
contains the new color table vectors and even the index of
the color table you just selected. The code looks something
like this:

```
theEvent = { XCOLORS_LOAD, $
        ID: widgetToNotify, $
        TOP: topLevelBase, $
```

```
        HANDLER: 0L, $
        R:r, G:g, B:b, INDEX:ctindex }
  Widget_Control, widgetToNotify, Send_Event=theEvent
```

The other program just deals with this event in the normal way it deals with *any* event. That is to say, the event handler for the widgetToNotify widget does whatever it is that it is suppose to do.

You can read more about this technique, including how you can extend the idea to objects, in this article on my web page:

http://www.dfanning.com/tips/using_xcolors.html

Cheers,

David

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