

---

Subject: Re: programming with widgets

Posted by [David Fanning](#) on Wed, 13 May 2009 00:58:39 GMT

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

---

nata writes:

```
> I'm a little bit confused when I use widgets..
> To create a widget, sometimes I use objects with a function called
> GenericClassEventHandler.
> Something like this:
> function widget::Init()
> self.base=widget_base(uvalue=self)
> ...
> xmanager, 'widget', self.base,
> event_handler='genericclassevent_handler'
> return,1
> end
>
> Sometimes I create procedures using the state structure in a User
> Value. For example:
> ...
> state={var_a: 0!, var_b: 0!}
> widget_control, base, set_uvalue=state, /no_copy
> ...
> I don't know which is the best option to preserve the application
> state and some variables...
>
> Now, I need to create a widget to display a list of options. The user
> will select one option and he'll close the widget. Something like the
> DIALOG_MESSAGE() using the QUESTION keyword.
> Something like this: result=MY_WIDGET(options)
>
> So, for do that, which is the best option?? Any ideas or
> suggestions ?
```

I'm a bit confused when I read your question. :-)

It seems to me what you want is a pop-up dialog widget, which just collects information from a user and returns it to the caller of the function. If that is the case, I don't think it much matters how you write it. Whichever way is faster for you, I guess.

But if you want to store the information somewhere, so it can be retrieved later (some kind of "configuration" object), then I think I would write this as an object with storage for the parameters of interest and a "GUI" method that could implement the pop-up dialog part of

the program.

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Sepore ma de ni thui. ("Perhaps thou speakest truth.")

---