
Subject: Re: Dynamically adding and removing widgets
Posted by peter.albert@gmx.de on Thu, 24 Nov 2005 07:16:36 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi all,

well, I'd guess that the state variable **does** exist in the routine "show_phenomenon":

In "pro event_handler", state is assigned to the uvalue of the parent widget :
WIDGET_CONTROL,Event.TOP,SET_UVALUE=state,/NO_COPY

It is correct that the variable "state" itself is undefined afterwards, but it does still exist as the UVALUE or the root widget. Then the routine "show_phenomenon" is called with the "event" variable as a parameter, which is fine, as "event" also still exists. Especially, event.top still points at the parent widget. Therefore, when calling

widget_control,Event.Top,GET_UVALUE=state

within "show_phenomenon", everything works as expected and the "state" is read from the parent widget's uvalue.

So where is the problem coming from?

The real problem is with the "base" variable:

You use

```
mask_disp = WIDGET_DRAW(base,xsize=numCols,ysize=numRows)
```

where "base" is expected to be a valid widget ID. As you showed yourself, however, "base" holds the string "mask_tab":

```
>> IDL> print, base  
>> mask_tab
```

Why?

Because in the line above you use

```
base = widget_info(state.mask_tab,/UNAME)
```

Using /UNAME with WIDGET_INFO returns the user name of the widget.

So probably

```
mask_disp = WIDGET_DRAW(state.mask_tab, xsize=numCols,ysize=numRows)
```

is just what you want?

It will create the new draw widget right in the widget whose ID is state.mask_tab.

Cheers,

Peter
