Subject: Re: xmanager does not block
Posted by davidf on Mon, 28 Feb 2000 08:00:00 GMT
View Forum Message <> Reply to Message

marc (m_schellens@hotmail.com) writes:

> I have a program, wich calls:
> xmanager,name,tlb
> (without /NO_BLOCK) but xmanager immediately returns.
> Is there a common mistake I might have made?

No mistake. I'm guessing you are calling this program
from within a widget program that is *itself* blocking.
Only the first program that calls for a block actually
blocks. All the rest run through the block. In practice
you usually make these other program modal programs.

You can read this article for more information:

http://www.dfanning.com/tips/modal_blocking.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

Subject: Re: xmanager does not block
Posted by Karri Kaksonen on Mon, 28 Feb 2000 08:00:00 GMT
View Forum Message <> Reply to Message

Marc,

I assume that you try to do the same thing I had problems with.
To have IDL do calculations while you want to access menus
and process keypresses etc.

I found a solution that works for me:

Create a procedure that sends events like

```
pro rawgui::send_event,widgetID
    event = { $
```

```
        id:widgetID, $
        top:self.tlb, $
        handler:widgetID, $
        select: 1l $
    }
    widget_control,widgetID,send_event=event
end
```

Create a procedure that starts the processing like

```
pro rawgui::reconstart
    self->send_event,self.reconrestID
end
```

Create a procedure that does one piece of processing

```
pro rawgui::reconstep
        self.proviewfileio->prodat
        if more_to_do ne 0 then self->send_event,self.reconrestID
end
```

Create an event handler like

```
pro rawgui_events, event
    widget_control, event.top, get_uvalue=self, /no_copy
    widget_control, event.id, get_uvalue=request
    case request of
    'readall': begin
            self->reconstart
    end
    'reconrest': begin
            self->reconstep
    end
end
```

Now you can put xmanager where it belongs

```
pro rawgui::makewindow
        self.viewID = obj_new('idlgrview', color=[80,80,80],
viewplane_rect=self.viewport)
        tlb = widget_base(title='Marconi Medical MRI Data Display',
column=2, tlb_size_events = 1, mbar=menuID)
        ... a lot of buttons etc ...
        self.reconrestID = widget_label(tlb, value='',
uvalue='reconrest', event_pro='rawgui_events')
        xmanager, 'rawgui', tlb, /no_block, group_leader=groupLeader,
event_handler='rawgui_events'
end
```

I also believe that xmanager should always return.
to block or not to block refers to whether you can
use the command line while your program executes.

--
Hope this helps...,

Karri

marc wrote:
>
> marc wrote:
>>
>> I have a program, wich calls:
>> xmanager,name,tlb
>> (without /NO_BLOCK) but xmanager immediately returns.
>> Is there a common mistake I might have made?
>>
>> thanks,
>> :-) marc
> Might it be because I call xmanager from within a widget procedure?

Subject: Re: xmanager does not block
Posted by marc schellens[1] on Mon, 28 Feb 2000 08:00:00 GMT
View Forum Message <> Reply to Message

marc wrote:
>
> I have a program, wich calls:
> xmanager,name,tlb
> (without /NO_BLOCK) but xmanager immediately returns.
> Is there a common mistake I might have made?
>
> thanks,
> :-) marc

Might it be because I call xmanager from within a widget procedure?