
Subject: Re: running an application from the IDL virtual machine

Posted by [Helder Marchetto](#) on Fri, 20 Nov 2015 11:26:35 GMT

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

On Friday, November 20, 2015 at 11:14:09 AM UTC+1, Helder wrote:

> On Friday, November 20, 2015 at 10:41:53 AM UTC+1, marka...@gmail.com wrote:

>> On Friday, November 20, 2015 at 1:52:20 AM UTC+1, Jim P wrote:

>>

>>> You will probably want your main widget event loop to be blocking if that's all your object is responsible for.

>>

>> I don't understand what you mean here. Do you mean that the object initialization should never finish, and that the variable foo in the above example should not be returned, until the application closes?

>>

>> The application itself is a graphics window. It can spawn further instances of itself if it needs to create new windows. The object initialization needs to finish smoothly, in order for things like that to work.

>

> Hi,

> the way I do this is to make an executable of a pro that calls the object.

> This is how it looks like:

>

> function myObj::init

> ;make your widget here or elsewhere within the object

> ;no need to block the widget

> self.wBase = widget_base()

> self.wlabel = widget_label(self.wBase, value='show something')

> widget_control, self.wBase, /realize

> xmanager, 'myObjWid', self.wBase, /no_block

> print, 'wid obj initialized'

> return,1

> end

>

> pro myObj__define

> void = {myObj, wBase:0!, wLabel:0!}

> end

>

> pro runMyObj

> obj = obj_new('myObj')

> print, 'obj called, exit pro. Obj lives on.'

> end

>

> This does the trick for me.

> Is this what you're looking for?

>

> Cheers,

> Helder

Hi,
just so to make sure FG also works, here is a minimal example. Once started, you can interact with the plot as usual.
What I always find strange, is that in my case the plot (widget_window) appears black until I run the mouse over the window...

But the rest works. Cheers, Helder

```
function myObj::init
;make your widget here or elsewhere within the object
;no need to block the widget
self.wBase = widget_base()
self.wWindow = widget_window(self.wBase)
widget_control, self.wBase, /realize
xmanager, 'myObjWid', self.wBase, /no_block
self.pp = plot(/test, current=self.wWindow)
print, 'wid obj initialized'
return,1
end
```

```
pro myObj__define
void ={myObj, wBase:0I, wWindow:0I, pp:obj_new()}}
end
```

```
pro runMyObj
obj = obj_new('myObj')
print, 'obj called, exit pro. Obj lives on.'
end
```
