Subject: Re: moving to objects / IDL objects and object graphics
Posted by David Fanning on Sat, 06 Oct 2007 23:48:13 GMT
View Forum Message <> Reply to Message

markb77@gmail.com writes:

> But could this be integrated with IDL object graphics? For instance,
> would the IDLgrView know what to do if I were to Add my Movie object
> to it? And would the IDLgrWindow know how to draw it?

If your object is the type of thing IDLgrView knows
what to do with (I.e, models, graphics atoms, etc.),
then IDLgrView will know what to do with it. Specifically,
it will need a DRAW method to implement the graphics
rendering. (Models, of course, *have* a draw method, so
if your new object is subclassed from a IDLgrModel object,
and I'm sure it will be, then the DRAW method is already there.)

> that's my question of the moment. I'm going to keep reading on this
> object stuff, but I'm wondering about integration with object
> graphics...

Be sure you write the DRAW method if you are creating new
graphics objects, or add your graphics objects to a model
(I.e, subclass your object from an IDLgrModel), and you
are good to go. See something like HCOLORBAR__DEFINE for
an example:

http://www.dfanning.com/programs/hcolorbar__define.pro

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.")

Subject: Re: moving to objects / IDL objects and object graphics
Posted by Robbie on Mon, 08 Oct 2007 22:33:44 GMT
View Forum Message <> Reply to Message

I think that you should only extend a IDLgrView object if you are
really creating a new graphics object. For example a colorbar is a
graphics object because it handles how to draw a colorbar and only
that.

A movie object would take care of getting data from a source and putting it on the screen. But, it also has to handle timer events, buffering and redrawing tasks.

Fields of such an object might be:
1) A widget base with the timer activated
2) A widget draw with GRAPHICS_LEVEL=2
3) The IDLgrView being drawn
4) The IDLgrImage which contains the frame buffer
5) A pointer or object reference to the actual movie data

The movie object would contain an event loop to handle events from these.
I would probably delegate 4 and 5 to a separate object again so that you can use various sources of movie data.
You could extend your movie object to include other widget buttons for controlling your movie.

Robbie