Subject: Puzzling over inherited objects
Posted by Helder Marchetto on Tue, 04 Jun 2013 21:51:47 GMT
View Forum Message <> Reply to Message

Hi,
I'm quite new to using objects and wanted to do a bit more with what I have been doing until now (not much anyway...). So here is the deal.
I want to create an object that incorporates (inherits?) the a typical object such as IDLanROI. So that the structure definition would look like this something like this:

class = {MyRoiObj, INHERITS IDLanROI, IDNr:0L}

where I simply added an ID to each IDLanROI. I still want to have all the functionality of the methods associated with IDLanROI.

The problem is: how would I assign the object (properties of) IDLanROI to my new object in the init function? I guess (and I'm quite sure) that something like
self = obj_new('IDLanROI')
or
void = obj_new('IDLanROI')
struct_assign, self, void

will not work. So the only solution I see at the moment is to create the structure as:

class = {MyRoiObj, IDLanROI_ref:OBJ_NEW(), IDNr:0L}

And then in the init or whereelse, assign the object as:
self.IDLanROI_ref = obj_new('IDLanROI')

So here is the question... is there a better way to do this? Ideally objects are made with inheritance... I simply can't find a way to use that in my case.

Any suggestions?

Thanks,
Helder