
Subject: Re: Commons, Was: can i place a job advert
Posted by [Richard Younger](#) on Tue, 27 Nov 2001 01:17:28 GMT
[View Forum Message](#) <> [Reply to Message](#)

"Pavel A. Romashkin" wrote:

- >
- > This topic is irresistible.
- > How about we issue a Challenge:
- >
- > Please modify the object definition below or create a method for objects
- > to be aware of each other after creation:

Let the testing begin!

Well, I can do it with a system variable (see below) but this has all of the disadvantages of the common block, including namespace issues. And the namespacing discussion has already been done. I could also do it with a few lines inserted at the beginning of the startup file, rushing to gain the ObjHeapVar1 space first. I could also conceive of making a DLM to declare and reference a global C variable.

But you've specifically said that there can be no direct context communication between objects. The problem is framed so that one needs some information on a global level. Really, if you're going to subscribe to objects fully, you shouldn't need to find other objects without referencing them.

The question is what are you doing that truly requires global information? Besides, um, main level interactivity? And besides, er, main level widget interaction? And, er, um, *mumble* hardware-dependent stuff, like *mumble* display controls. Okay, so maybe there occasionally are good reasons to need some global information. :-)

Sorry for sounding grumpy, and I agree that commons have their place, I'd just thought I'd chime in with something a little more reasonable than "Commons are bad."

Best,
Rich

--

Richard Younger

FUNCTION Test::init

CATCH, error

```
IF error NE 0 THEN BEGIN
  catch, /cancel
  table = OBJ_New('IDL_container')
  DEFSYSV, '!RYtable', table
ENDIF

table = !RYtable
table->add, self

RETURN, 1

END

PRO Test::cleanup

table = !RYtable
table->remove, self
IF table->count() EQ 0 THEN BEGIN
  Obj_destroy, table
  !RYtable = OBJ_NEW()
ENDIF

self->IDLgrModel::cleanup

END

FUNCTION Test::get_others

table = !RYtable
RETURN, table->Get(/all)

END

PRO Test__define
self = {TEST, $
  data_holder : ptr_new() , $
  inherits IDLgrModel $
}
END
```
