Subject: Re: Global variables and IDL
Posted by roberson on Tue, 13 Apr 1999 07:00:00 GMT
View Forum Message <> Reply to Message

In article <19990412230259.15932.00002313@ng117.aol.com>,
Russ051990 <russ051990@aol.com> wrote:
:I'm certain that IDL has a global variable (I remember it being mentioned in
:Fanning's book), but don't remember how to invoke it. Maybe someone else can
:provide that for you.

Use a common block. That requires adding only one extra statement to each routine that uses an element of the common, and requires no other code changes.

:In the meantime, you could use a "FORTRAN-like" structure to hold your "global"
:variables, and pass just the pointer to the structure around if you'd like.
:The structure can be declared like this
:common = PTR_NEW ({ varname1:value1, varname2:value2, ..., varnamen:valuen})

You would have to carry the pointer around everywhere (the pointer name will not be global in scope even though the heap values pointed to are global); and you would have to make many code changes to dereference. Fortunately those changes could be made in batch mode with any good editor.

The online help shows, in Widget Example 2, another method. When you are using a widget and you ask for the value of the 'top' field of the event structure, you get the user value associated with the top level widget of the widget you are processing the event for. If that user value has been set to a widget ID, you can use WIDGET_CONTROL routines to extract values from that widget, or to set values in that widget. But if you're going to do that and you don't need the graphical part to change, you might as well make the user value a PTR. [Not a structure, though, as it is a copy of the value that is put into the widget unless you use /NO_COPY.]

In any case, this discussion is inappropriate for comp.lang.idl (which is about the Interface Definition Language). I have cross-posted to the appropriate newsgroup. comp.lang.idl-pvwave, and set followups to go to there.