Subject: Re: Global variables and IDL
Posted by rmlongfield on Wed, 14 Apr 1999 07:00:00 GMT
View Forum Message <> Reply to Message

HI All, I've been thinking about this question myself lately. As far
as I understand, the use of COMMON blocks is not recommended when working
with widgets, but is ok otherwise. I have heard that there may be a problem
if there are too many variables, but 20 seems ok to me.

I've got another question regarding this issue. I've been using IDL in
widget p rogramming. Following DWF's advice, I do not use COMMON statements.
  Here comes a big HOWEVER, however. I have an image processing tool which
uses a base data set but many different widget modules, each with its own TOP
LEVEL WIDGET. I would like to have access to this data from whatever or
whichever widget I am working in. The data sets I read are created in the
middle of my processing, so it would do no good to read it at the start and
have a general pointer that can be included in each Top Level Widget. The
only alternative is to keep track of all the widget ID's and then notify them
when the data has been updated, a procedure discussed in DWF's book.

My problem is that this tool of mine is getting rather complicated (read:
messy) and I have three WIDGET_CONTROL, GET_UVALUE statements at the
beginning of many of my event handlers, because I have several info pointers
which hold information that I need. (One each from the event ID, top and main
info) Maybe I am just dis-organized, but it seems to be getting out of hand.

 I am wondering whether using a COMMON statement for a set of
variables or data arrays that aren't changing would not be such a bad idea.
What about using constants such as PI or earth radius? Where can they be
defined once and then used throughout a program ? I am considering putting
my data in a COMMON statement so that I can get to it in an easy way (i.e.
without (*(*infoPtr).dataArrPtr) type statements all over the place)

It is this or re-writing the tool from scratch (which it really needs if only
I could find the time).

Rose