Subject: Re: Discussion on global variables in IDL - reply to Martin and Pavel
Posted by alt on Thu, 02 Aug 2001 12:00:38 GMT
View Forum Message <> Reply to Message

First of all I'd like to thank you for information that using COMMONs
is not only not cool but could be dangerous for a life. I'll
take this into account in the future :-)

I agree with you on the whole and want just clarify some items of my
subjective position.

The fact is that very often you don't know exactly what you are
going to get from program. And if you write in C (even more with
object programming) you really have to design all data structure
before writing. And it's very tempting not to do so. And
I've created illusion for myself that IDL allows me to just sit
and program. At least I am not disappointed in this yet. If I know
exactly what I'm going to do why should I use IDL. I would
better write in C (for ex.) using math libraries and I get fast cheap
executable program with high flexible GUI. As many do by the way. The
fact that if I don't know exactly what I'm going to do I
spend five times much more time on that in comparison with IDL. And I
should be much more qualified in programming. Of course, it's
right mainly for scientific data processing purpose.

Once I've met an idea that a book on mathematics can be either
exact or understandable. I seem the same is right for programming to
some extent. Program in IDL is not exact but understandable. IDL
implicitly takes upon itself many of the functions that I don't
want think about during data processing. I don't need declare
variables, do type conversion, write FORs to copy arrays, reserve and
clear memory and so on. I can interactively run simple debug code to
found out what's happened on error. It is excellent. And I
dislike when I should do something that doesn't concern to the
point of the process. I must not use pointers if I don't use
dynamic data arrays. I must not use structures if I don't have
actually structured data. And I must not use object-oriented technique
if I don't need inheritance for example. I don't know
really if I succeeded to put into words my conception but I have
tried.

Cheers,
Dmitriy

Remote Sensing Center of ISTP, Irkutsk
alt@iszf.irk.ru