Subject: Re: saving variables between calls to a procedure?
Posted by Reimar Bauer on Wed, 31 Jul 2002 19:08:07 GMT
View Forum Message <> Reply to Message

Shawn wrote:

> Hello,
> I am just curious if there is a better way to save variables
> between a call to a procedure than to define them in a common block
> which is shared between procedures, or passing them back and forth.
> One person suggested to me saving them to a file and re-reading a
> file. But I am hoping for some simple way to declare a particular
> variable as a "saved" variable, which IDL will remember the next time
> the procedure is called. It actually just occured to me that I might
> declare a common block which is actually not shared with any other
> procedure, then I would be able to avoid mistakenly using the variable
> in that procedure in a way that I did not intend to use it. Is it
> possible to do this, declare a common block, but not use it in any
> other procedure, and will IDL remember the variables if the common
> block is not used in any other procedure? Of course it is simple for
> me to test this on my own, so I guess I am really still asking if
> there is a better way.
> Thank you,
> Shawn Young

Dear Shawn,

I don't like common blocks too. Most times I am using parameters
as variable or structure or pointer.
It depends on what's the routine should do.

For example you can define a pointer as heap var or with a value
outside in your main routine.
If you know the reference ptr then you can access from
all routines you want the data behind this pointer.

```
pro define,ptr
 *ptr=[*ptr,10]
end

IDL> ptr=ptr_new(0)
IDL> define,ptr
IDL> PRINT,*ptr
IDL> 0 ,10
```

Reimar

.