
Subject: Re: Pass by value and performance
Posted by [Paul Van Delst\[1\]](#) on Wed, 14 Dec 2005 17:02:14 GMT
[View Forum Message](#) <> [Reply to Message](#)

David Fanning wrote:

> Kenneth P. Bowman writes:

>

>

>> True, and I considered that possibility. It would work fine for passing
>> the arrays amongst my own routines, but I have to dereference the
>> pointer to pass the array itself into system routines. Dereferencing a
>> pointer is an expression, is it not?, and expressions are passed by
>> value.

>

>

> Ah, you would be interested to read the new pointer
> section of my book that I just completely yesterday!
> (Well, actually, you could read the damn pointer tutorial
> where I learned about this, too.) Here is the news that
> even I didn't fully appreciate until I put it down in my
> own words. Pointers are just regular IDL variables!!

In the context of this thread (and your example below), shouldn't one say that
/dereferenced/ pointers are just regular IDL variables (i.e. they're not expressions) ??

I don't think I'm splitting hairs here (or am I?)

>

> Consider this function:

>

> PRO JUNK, var

> var = var * 5

> END

>

> And do this:

>

> IDL> a = Ptr_New(5)

> IDL> junk, *a

>

> Here is the question. What is the result of this command?

>

> IDL> Print, *a

>

> The answer might surprise you! :-)

After my recent grokking of objects (thanks to that "object for dummies" website posted
earlier this month), the answer didn't surprise me as much it would of, but it's still a
good reinforcement.

It makes me wonder though - have IDL pointers always behaved that way, or is it a moving target (ala relaxed structure definitions post IDL 5.3 talked about in another thread) ?

paulv

--

Paul van Delst
CIMSS @ NOAA/NCEP/EMC
