
Subject: Re: pointer question

Posted by [Martin Schultz](#) on Mon, 26 Mar 2001 09:05:13 GMT

[View Forum Message](#) <> [Reply to Message](#)

Ted Graves wrote:

>
> Hi all,
>
> Another lurker question ... let's say you define a pointer using the PTR_NEW
> function and assign to a variable x. As long as you keep track of x and don't
> reassign x and lose the pointer to the heap variable, things are great. You
> can remove the heap variable from memory using the PTR_FREE procedure.
>
> But now let's say i have a function TEST that takes a pointer as an argument,
> and i want to create a pointer on the fly to use in TEST. So i do something
> like
>
> result = TEST(PTR_NEW(value))
>
> where value is whatever i want the heap variable to be. What happens to the
> heap variable assigned in this statement after TEST returns? I'm assuming
> from that because of the way it was created, a heap variable now exists that i
> can't easily get rid of without using HEAP_GC.
>
> Me and my sloppy programming ...
>
> Ted Graves
> Magnetic Resonance Science Center, UCSF

Let me second Paul here. WHY? This is the real question here. As I
relearned only recently, IDL automatically passes variables by
reference unless you index them (or do whatever other weird things to
them). So from a program efficiency standpoint, you are passing a
pointer when you simply write

```
result = test(value)
```

Now if you really want to give test only the data of the first column
(or row or whatever), you really should, as Paul suggests, create and
destroy the pointer in the caller routine, i.e.:

```
x = Ptr_New(value[0,*])  
result = test(x)  
Ptr_Free, x
```

But even then: you don't even need a pointer here! The following is
exactly the same in terms of efficiency, and it doesn't require a
cleanup (if you write it inside a procedure or function).

```
x = value[0,*]
result = test(x)
```

So, why use pointers at all, the witty lurker might ask now? Let me dare to say that you only need them within structures (or objects for that purpose), i.e. if you need to access a variable at a certain place but you don't know it's shape or size beforehand.

Cheers,

Martin

```
--
[[ Dr. Martin Schultz  Max-Planck-Institut fuer Meteorologie  [[
[[          Bundesstr. 55, 20146 Hamburg          [[
[[          phone: +49 40 41173-308          [[
[[          fax: +49 40 41173-298          [[
[[ martin.schultz@dkrz.de          [[
[[ Dr. Martin Schultz  Max-Planck-Institut fuer Meteorologie  [[
```