Subject: Re: taming the shrew, a.k.a. structure
Posted by david[2] on Tue, 31 Jul 2001 23:26:49 GMT
View Forum Message <> Reply to Message

Todd Clements writes:

> Pointers
> are fun and useful things, but that also means that you have to worry
> about cleaning them up when you're done.
>
> myStruct = {myStruct, array1: ptr_new()}
>
> Then, in your code:
>
> myStruct.array1 = ptr_new(fltarr(startSize))
>

Too true. Although one could easily write a little
program to free structure pointers:

```
PRO Free_Pointers, structure
FOR j=0,N_Tags(structure)-1 DO BEGIN
   type = Size(structure.(j), /TName)
   IF type EQ 'POINTER' THEN Ptr_Free, structure.(j)
ENDFOR
END
```

Extra credit for making this recursive. :-)

> Of course, if you need to shorten or lengthen this later, you have to
> remember to dispose of the pointer that you made AFTER you make the new
> one.

Not really. The nice thing about IDL pointers is that
IDL will take care of all the memory manipulation
for you. You don't have to worry about it at all.

> temp = myStruct.array1
> myStruct.array1 = ptr_new((*myStruct.array1)[0:1024])
> ptr_free, temp

This really becomes nothing more than this:

```
*myStruct.array1 = (*myStruct.array1)[0:1024]
```

There is no need to free the old pointer, make
a new one etc. Pointers are like IDL variables
in this respect. They *always* point to the

current thing you have pointed them too.

> It's sometimes a lot of work to use pointers, but they do exactly what
> you describe you want to.

If by "a lot of work" you mean you have to use
more parentheses than normal, I would agree with
you. Sometimes that syntax drives me crazy! But
the benefits you gain far exceed the cost.

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155