Subject: Re: IDL Memory Leaks
Posted by John-David T. Smith on Mon, 05 Nov 2001 18:24:53 GMT
View Forum Message <> Reply to Message

Myron Brown wrote:
>
> Recently, I have noticed that my IDL programs leak memory, but I never
> use pointers directly. This is true when running with IDL on a Windows
> PC or on an SGI workstation. Widgets seem to be one source of
> problems. File I/O seems to be another, but I'm not yet sure. Due to
> the problems I'm having with memory leaks, my long runs eventually die
> when memory is exhausted.
>
> Does anyone have any hints on ways to avoid memory leaks in IDL?
>
> Please reply to my e-mail address, since I don't often use newsgroups.
>
> Thanks.

Myron:

Side point: comp.lang.idl is an altogether different newsgroup, unrelated to RSI's IDL. Interface Definition Language, or some such. Just FYI, because I'm sure they're as confused by our posts as we are by theirs.

I can suggest a few places to look:

1. Pointers. Are you *sure* the widgets you use aren't compound widgets with pointers in them, or that all your I/O routines are pointer-free? You can, at any point, find out what's on the object/pointer heap by:

IDL> help,/heap

If anything is on the heap, look for dangling pointer variables with:

IDL> heap_gc,/verbose

which will tell you *which* pointers were causing memory leaks. Repeating this a few times can help you zero in on the culprit.

2. Widget's with large UVALUEs inside. Widgets are in many ways just like pointers: they live on a global heap, are referenced by a unique ID, and can point to not one, but several different values of different sizes. This is perhaps why David Fanning used empty base widget's UVALUEs as pointers in the good old days before "real" pointers were introduced by RSI (possibly prompted by the embarrassment he was causing

them ;). If you're sticking very large things (like arrays) inside widget UVALUE's, and not properly destroying these widgets, you'll be in exactly the same boat as with pointer leaks.

Unfortunately, unlike pointers, there's not a nice way of investigating what's on the "widget heap" at a given time, that I know of. You can use the help,/MEMORY command to try to isolate big jumps, and even to analyze the size being added. If it's close to an array size you're sticking in a widget, then that's a very strong clue.

Good luck,

JD
