Pavel Romashkin (promashkin@cmdl.noaa.gov) writes:

> I have just lost an hour and a half trying to find a heap leak in my
> code. It is a simple object graphics, universal replacement for PLOT but
> with zoom, edit etc. capabilities. Anyway, once I kill the plot window,
> a cleanup routine is called that does total sweep of the heap variables
> by looping through object fields and destroying objects and pointers.
> Once its done, I'd call our favorite Heap_gc to see if there was a leak.
> Sure enough, it was always finding and deleting an instance of
> IDLGRFONT. I tried everything I could: commented out the only line where
> IDLGRFONT was used, tried all sorts of explicit object destruction
> immediately after creation, etc. I felt it was not my fault, so I tried
> "Clear IDL" and .Reset, and .Full_reset. Same thing: once I recompile,
> run and kill the object tree, IDLGRFONT would still always be found by
> Heap_gc. It was not a fake, as its heap ID would be always higher each time.
> Frustrated, I quit IDL and restarted. What do you think? It stopped
> doing that. I swear the code is the same, because I went through every
> single line checking it, saved it, tried it, got a leak and quit IDL.
> But after restart IDLGRFONT stopped appearing in the heap.
> I have no idea what it was, but it certainly complicated my leak search.

Etc., etc., etc.

> Does anybody have an explanation?

Yes, but I'm saving it for the IDL EPA exam. :-)

Cheers,

David

P.S. Let's just say now would be a good time to put
on that Lauridsen Lux Aeterna CD and light a candle. :-)

--
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