Subject: Re: how does /no_copy work???
Posted by davidf on Thu, 03 Jun 1999 07:00:00 GMT
View Forum Message <> Reply to Message

John Persing (persing@frii.com) writes:

> But let me ask, how can this be possible when deal with a variable that
> "starts" on the stack and "ends up" on the heap? If B is an ordinary array
> and A is property of an object, then this is what will occur. The heap and
> stack are entirely different memory locations.

I'm rapidly getting out of my depth here, but it seems to me that
the *object* itself is on the heap, but that the actual data that
fields in the object point to can be anywhere in process memory.
All that has to be stored in the object field is a pointer
(a *real* pointer, not an IDL pointer) to the real data. This
is what is passed, isn't it, when a variable is passed by
reference? If that wasn't the case, how else could a variable
be stored in a widget user value with NO_COPY, which to my
mind is equivalent to the heap (I.e, a global memory location)?

And keep in mind that "stack" and "heap" have meanings in IDL
that *may* not correspond to what you usually think about when you
use these terms.

Whew, I can't feel the bottom any more! :-(

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