Michael Wallace writes:

> IDL is vague! IDL is scary! While I love the loose typing for
> procedural programs, I can't stand it for objects. I must admit that
> even though I haven't programmed with objects in IDL, I have written
> several object graphics programs, and each time I write one I cringe a
> little because of what IDL calls an "object." Using them is so
> backwards in some respects. Maybe this is a wrong impression, but the
> more I learn, the more so-called IDL "objects" appear like "glorified
> structs" or maybe just "some stuff thrown together."

Well, this is the literal truth when something gets to be
at least 20 years old. But isn't it amazing that a language
like IDL can exist for that long? It is probably a miracle
that it is not messier than it is.

> The OO programmer in me really wants to argue the comment of IDL having
> "perfect polymorphism," but I'll remain civil. Gotta remember I'm on
> different turf than normal in this newsgroup.   ;-)

Well, most of the IDL object programmers I know don't
even know what polymorphism means (I admit I don't), so
"perfect polymorphism" is just something that trips lightly
off the tongue. Yes, it is messy and awful and "some stuff
thrown together", but it is also so much FUN! Can't really
say that about JAVA, I don't think. ;-)

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/