

Reinhold Schaaf (Kakadu42@gmx.net) writes:

- > As a second point, one
- > really has to get used to the fact that objects cannot be created on the
- > stack, (stack objects are destructed automatically on exit from the
- > program unit in which the object was constructed). Hence one is forced
- > to destruct any object manually, which is highly error-prone and makes
- > life really uncomfortable.

I am finding this to be the case too. Having learned my lesson with the object graphics programs I naively offered up as "examples", I am trying to be extra careful with memory management. But I currently have six pointers to objects hanging around and I will tell you they are damned hard to track down! There must be a better way, or at least some way to get more information about them besides sprinkling a thousand Print statements over your code.

- > A third remark concerns event handling: It is not possible to define a
- > method of a class as the event handler of a widget.
- > As a consequence, one is forced to make the event handler a global
- > function. But global functions cannot access members of objects, so one
- > has to add methods to the class which would be unnecessary otherwise.

I have grappled with this, too. Having all of a sudden become enamored with object programming, it is not much of a leap to believe that this is the way widget programs should work. Especially compound widgets. Like Reinhold, I've found imperfect, interim solutions.

- > Things could be ways simpler if global functions were allowed to be
- > event handling routines! I wonder whether this could be changed in
- > future versions of IDL.

I haven't specifically discussed this with the folks at RSI (although I plan to), but I think OOP is probably in the same state that widgets were in when they were first introduced. That is to say, the first effort was pretty good, but nowhere near where they had to get to eventually.

As we gain experience with object programming I think we need to let RSI know what works and what can be done better with these objects. (J.D. Smith has already offered a number of excellent suggestions.) I really do think RSI is listening to customers in a more focused way than in the past. This

could be the time to really take IDL several steps forward
as a language.

Cheers,

David

David Fanning, Ph.D.
Fanning Software Consulting
E-Mail: davidf@dfanning.com
Phone: 970-221-0438
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
