Subject: Re: Writing an IDL application using a C++ data model.
Posted by David Fanning on Mon, 12 Nov 2007 20:09:45 GMT
View Forum Message <> Reply to Message

Robbie writes:

> I think why bother with a structure at all? In fact I am currently
> using an array of PTRs and array of strings because it is easier and
> probably just as fast.
>
> struct = {MY_OBJECT,$
> identifiers: ptr_new(), $
> values: ptr_new() $
> }
>
> At this point I wonder if I'm departing from a sensible usage of IDL.
>
> I think IDL is a bit slow with method calls, particularily when I'm
> looking for a child object with a particular property. A typical speed
> up trick is to use a common variable as the backing store for that
> property and search it using WHERE()

Here's my point. The '65 Mustang is cool and all that,
but you don't have to file the points and clean the carburetor
on the new model to get good performance. It just happens.

Why all the tricks? Why not just use a modern programming
language and be done with it? What is it about IDL that
keeps you so preoccupied with modifying it in all these
different ways?

Surely there is something better than IDL that meets your
needs. Or, isn't there?

I'm not trying to be antagonistic. I'm just curious.

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")