
Subject: Re: Object Madness or Restoring Nightmares

Posted by [tam](#) on Thu, 04 Mar 2004 14:58:52 GMT

[View Forum Message](#) <> [Reply to Message](#)

David Fanning wrote:

> David Fanning writes:

...>

> Ah, here is the thing about this hierarchy that you
> should know. This is an object *containment* hierarchy.

> The top-level object is a container that holds all the
> other objects. Every object (except the top object) is
> both contained in a container and can (potentially)
> contain other objects. (All objects in my system
> inherit IDL_CONTAINER.)

>

> If you pick any object whatsoever out of this web,
> you can (apparently easily to judge from how fast
> IDL does it) traverse the entire object hierarchy.
> I can see that this is the reason IDL *must* save
> everything when I save even a single object that
> belongs in the hierarchy.

>

> What I can't see at the moment is a way out of
> this mess.

>

> Cheers,

>

> David

Hi David,

Forgive me if I'm asking stupid questions... (OK the if is superfluous!)

Clearly each object contains pointers to all of its children
so if you save the parent all the objects contained in it
are saved. But I don't see why a child (still talking about the containment
hierarchy, not the inheritance tree) needs to point to its
parent? Where is that pointer coming from and what is it doing?

I gather that each object needs to point to the class definition of the
top level container since that's also the class definition of
the root of the inheritance tree, but I wouldn't have thought
that saving the definition of the class means that you
have to save every instance of the class. That would certainly
seem like a broken implementation for the SAVE functionality.

Regards,
Tom McGlynn
