Subject: Re: Urgent object question
Posted by David Burridge on Thu, 24 Jan 2002 15:16:09 GMT
View Forum Message <> Reply to Message

Hi Ted,

Ted Cary wrote in message ...
>
> "David Fanning" <david@dfanning.com> wrote in message >
>
>> After a solid week of this we find that our ideas
>> *always* look wonderful about midnight. But not all
>> of them have survived the bright glare of morning. :-(
>> ... We are in the process
>> now of building a complicated enough system to test
>> whether it works or not.
>
> Morning is almost over, and it still seems to be a good idea. If you want
a
> not-so-complicated system to test, I can send you guys mine. I'm sure the
> boss would be happier if *real* programmers wrote my application.
Hopefully
> you and your friends will publish the results on your web page, so that I
> will know what I've done wrong. Who's the "we" you speak of anyway--I hope
> you haven't started referring to yourself and your computer in the first
> person plural...
>

I just got back from that week of programming Dave mentioned, so forgive the
delay in sticking my oar in!

The way I see it, using the IDL_Container (or a subclass if you need some
specific
behaviour) has a ton of advantages. The only problem with it is 1) you need
to be
absolutely clear on the ISA vs HASA relationship and 2) you can have get/set
calls
flying everywhere - causing infinite loops and making traceback impossible!

By only passing unresolved get/set keyword requests to the parent object,
we've cut
down on the tracking problem and eliminated infinite loops. This is
relatively easy
using the keyword inheritance methods. Secondly, we've buried all the
mechanics
in a single superclass (e.g. inheriting IDL_Container and passing unresolved
keywords to parent container objects) so that the hierarchy can be created
almost

transparently by simply inheriting our top-level object. Last of all, by accepting the
parent class as a param to the init method, our object adds ITSELF to the container,
adding to the transparency effect.

As Dave said, this idea is still bedding down. I wrote a little test app which proved the
concept, now we're trying to break it!-) Hope this garble is understandable and helpful!

Dave
