
Subject: Re: What about real polymorphism ??

Posted by [Mark Hadfield](#) on Thu, 09 Dec 2004 23:00:31 GMT

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

David Fanning wrote:

> Michael Wallace writes:

>

>

>> Quite a thread you guys have here. Anyway, I can't speak for objects in
>> IDL since I haven't actually learned how to use them yet, although I
>> keep planning to. I know exactly what Antonio is saying, so I'm going
>> to try and give another example of how things work in Java. Maybe
>> seeing this will help you IDL folks to better understand the Java side
>> of the question. I'm going to be using Java code in my example, so I
>> hope you can follow it. ;-)

>

>

> Thank you for trying to shed light on this, Michael.
> I can read your code well enough. What I can't follow
> is why Antonio thinks this can't be done in IDL. :-)

I think Antonio's confusion came from his experience with languages like C++ and Java, which have a different object model from IDL. In C++ and Java, the fact that two classes have a method called Draw does not imply these methods are related. For them to be related, the classes have to have a common superclass which itself has a Draw method. The superclass's Draw method will often be virtual, ie. not implemented, but it has to exist.

Then if you want to put objects of different classes in a container, they all have to share a common superclass and when you put them into the container they kind-of-lose their identity (sorry for the technical terms here) So when you take an object back you have to cast it back its superclass before you can call its Draw method.

I hope nobody who knows anything about C++ and/or Java is reading this!

Or something like that. There are entire tomes devoted to this stuff. The thing is that in IDL you don't have to worry about all this and this can be confusing for people who learn about the more strictly-type languages first.

> So, in Antonio's case, if he wants to treat his MEN
> and WOMEN objects as "people", the more power to him.
> Anyone who interacts with one of his "people" is going
> to find the proper method called without him having to
> do anything extra about it. That seems like perfect
> polymorphism to me. :-)

Me too, but the strictly-typed people find it all a bit vague & scary.

--

Mark Hadfield "Ka puwaha te tai nei, Hoesa tatou"

m.hadfield@niwa.co.nz

National Institute for Water and Atmospheric Research (NIWA)
