Subject: Re: Object-Oriented Programming Question
Posted by davidf on Wed, 17 Dec 1997 08:00:00 GMT
View Forum Message <> Reply to Message

Peter Stoltz (pstoltz@pppl.gov) writes:

> I have finally gotten around to trying out object-oriented programming
> with IDL, and I have a question about data encapsulation.
>
> I define an object structure A that has as one of its data members
> another object B (A has a B). So far as I can tell, when one creates
> an instance of A, one cannot invoke the methods of class B through
> the syntax
>
> IDL> a=obj_new('A')
> IDL> a.b->some_method

I think the correct syntax is this:

```
a->b::some_method
```

This will search the superclass b and any of its superclasses.

> Also, is there a mailing list or anything specifically about
> object-oriented programming in IDL?

This is a good idea. I've been thinking about some kind of
programming newsletter. Maybe when I have some time...

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/

Subject: Re: Object-Oriented Programming Question
Posted by rivers on Thu, 18 Dec 1997 08:00:00 GMT
View Forum Message <> Reply to Message

In article <34986079.BC1BBB4F@pppl.gov>, Peter Stoltz <pstoltz@pppl.gov> writes:
>> Hi everyone-
>

> I have finally gotten around to trying out object-oriented programming
> with IDL, and I have a question about data encapsulation.
>
> I define an object structure A that has as one of its data members
> another object B (A has a B). So far as I can tell, when one creates an
>
> instance of A, one cannot invoke the methods of class B through the
> syntax
>
> IDL> a=obj_new('A')
> IDL> a.b->some_method
>
> % Object instance data is not visible outside class methods
> % Execution halted

It's much simpler:

IDL> a->some_method

will invoke member b's "some_method". This works fine as long as "some_method" is not "overloaded", i.e. "a" does not have a routine of the same name. If it does, then you will get "a's" version by default. There is a syntax which will allow access to b's method even in this case. I forget it, but it's in the manual.

Mark Rivers (773) 702-2279 (office)
CARS (773) 702-9951 (secretary)
Univ. of Chicago (773) 702-5454 (FAX)
5640 S. Ellis Ave. (708) 922-0499 (home)
Chicago, IL 60637 rivers@cars.uchicago.edu (e-mail)

or:
Argonne National Laboratory (630) 252-0422 (office)
Building 434A (630) 252-0405 (lab)
9700 South Cass Avenue (630) 252-1713 (beamline)
Argonne, IL 60439 (630) 252-0443 (FAX)