Subject: Re: Object Funkiness
Posted by J.D. Smith on Wed, 30 Jul 1997 07:00:00 GMT
View Forum Message <> Reply to Message

I just realized that this isn't actually a good test of dynamic method binding. So I made a and b *different* kinds of objects, each with a Message method. The same problem persists, with the same properties.

JD

> P.S. Except for this problem, dynamic binding seems to work well.

This is verified.

Subject: Re: Object Funkiness
Posted by J.D. Smith on Thu, 31 Jul 1997 07:00:00 GMT
View Forum Message <> Reply to Message

J.D. Smith wrote:
>
> OK OOP experts. Here's a conundrum...
>
> here's a procedure to test dynamic binding of methods in IDL...
>
> pro testdb
> a=obj_new('eeke','a')
> b=obj_new('eeke','b')
> list=[a,b]
> ran=fix(randomu(sd) ge .5)
> list[ran]->Message, ran, 1-ran
> obj_destroy,list
> return
> end
>
> and the file eek__define.pro contains:
>
> pro eeke::Message, ran, ran2
> print,self.val,' random: ',ran, ran2
> return
> end
> function eeke::Init,val
> self.val=val
> return,1
> end
> pro eeke__define
> struct={eeke,val:''}
> return

> end
>
> When I run it I get
>
> IDL> testdb
>
> list[ran]->Message, ran, 1-ran
> ^
> % MESSAGE: Incorrect number of arguments.
> At: /u/jdsmith/idl/pro/mylib/testdb.pro, Line 6
> % Compiled module: TESTDB.
> % Attempt to call undefined procedure/function: 'TESTDB'.
> % Execution halted at: $MAIN$
>
> If I change the name of Message to something else, or take away the
> arguments to message in both the method definition and the call, the
> error disappears. Is 'message' somehow different than other names? I
> thought it could be shadowing the idl built-in message, and screwing up
> the dynamic binding, but I've tried tons of other built-in names (like
> 'print', 'xmanager', etc.) with nary a problem. Perhaps 'Message' is
> some IDL internal method for all objects... anyway it is very
> troublesome, and I'd like to figure out the cause. Any help would be
> greatly appreciated.
>
> JD
>
> P.S. Except for this problem, dynamic binding seems to work well.

I've found that this isn't limited to dynamic binding. Indeed, if you
just try:

a=obj_new('eek','aval')

and then, e.g.:

a->Message,1,0

you will still get the error. I've alerted the tech support crew at
RSI. At least a list of which method names we're not allowed to use
would be good. But, really, this bug needs resolution.

JD

Subject: Re: Object Funkiness

Posted by Phil Williams on Fri, 01 Aug 1997 07:00:00 GMT
View Forum Message <> Reply to Message

J.D. Smith wrote:
>
> J.D. Smith wrote:
>>
>> OK OOP experts. Here's a conundrum...
>>
>> here's a procedure to test dynamic binding of methods in IDL...
>>
>> pro testdb
>> a=obj_new('eeke','a')
>> b=obj_new('eeke','b')
>> list=[a,b]
>> ran=fix(randomu(sd) ge .5)
>> list[ran]->Message, ran, 1-ran
>> obj_destroy,list
>> return
>> end
>>
>> and the file eek__define.pro contains:
>>
>> pro eeke::Message, ran, ran2
>> print,self.val,' random: ',ran, ran2
>> return
>> end
>> function eeke::Init,val
>> self.val=val
>> return,1
>> end
>> pro eeke__define
^^^^^^^^^^^^^^^^^

The __define method should be a FUNCTION. I'm sure once this is done
everything should work fine.

>> struct={eeke,val:''}
>> return
>> end
>>

phil

--
/** ********/
Phil Williams, Ph.D.
Research Instructor
Children's Hospital Medical Center "One man gathers what
Imaging Research Center another man spills..."

3333 Burnet Ave.
Cincinnati, OH 45229
email: williams@irc.chmcc.org
URL: http://scuttle.chmcc.org/~williams/
-The Grateful Dead
/** ********/