

---

Subject: Re: keyword inheritance and object inheritance

Posted by [Ted Cary](#) on Sat, 23 Mar 2002 17:44:06 GMT

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

---

Hi Ben,

You should use `_REF_EXTRA` when you have output keywords. It should therefore not be necessary in an object's `SETPROPERTY` method, but is almost indispensable in any subclass object's `GETPROPERTY` method.

Output keywords are possible exactly because IDL functions are normally passed keyword arguments by reference, so that changing the value of a keyword argument inside a function will change the value of the argument outside the function as well. When keywords are stored and passed along in the `_EXTRA` structures created in function declarations, however, they are stored in fields by value and the reference information is lost. This is fine if you only want to `USE` the value, but not if you want to `CHANGE` the value stored in the argument. In order to change the value in some other function, `_REF_EXTRA` allows you to pass along the output keywords by reference.

Use `_REF_EXTRA` instead of `_EXTRA` in `GETPROPERTY` function declarations of subclass 'child' objects (Mother/Father, MyObj/YourObj in your example) if you want to override the `GETPROPERTY` methods of their superclass 'parent' objects. In the call to a superclass (parent's) `GETPROPERTY`, pass the structure stored by `_REF_EXTRA` via the `_EXTRA` keyword. Your `MOTHER::GETPROPERTY` is written correctly.

To correct your code rewrite all subclass `GETPROPERTY` methods so that they are like your `MOTHER::GETPROPERTY`. You can remove the `_REF_EXTRAS` from all the `SETPROPERTIES`.

Another option which avoids function overriding and `_REF_EXTRAS` is to use the `EXTRACT` function on David Fanning's website. It takes advantage of the fact that self fields of subclass objects inherit the fields of their superclasses. This will work if your self fields `ARE` the properties you want to extract, as in your example. For more complicated objects, however, `GETPROPERTY` is better. HTH

---

Subject: Re: keyword inheritance and object inheritance

Posted by [David Fanning](#) on Sat, 23 Mar 2002 20:45:05 GMT

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

---

Ben Tupper (btupper@bigelow.org) writes:

> I really don't understand how to retrieve object properties when it  
> comes to multiple object inheritance.

I'm pretty sure you are not alone, Ben. I *\*thought\** I understood it, but I spent two hours getting your example to work. Although, as it happened, I had the problem solved in five minutes, but I was mistyping "grandma" as "gra\*m\*dma". No wonder it was always undefined. Typical, I'm afraid. :-)

Anyway, after the jolt of Starbuck's put me onto the right track, I think I really do understand it now. At least I tried every single one of the perturbations at least 5 times. :-)

Here is the rule of thumb I would use. On every GetProperty method, I would *\*define\** a `_Ref_Extra` keyword. But on every *\*call\** to a GetProperty method I would use an `_Extra` keyword to pass the extra structure.

For example, here is how you wrote the GetProperty methods for your "mine" example:

```
;-----
; Getproperty
;-----
PRO grandmother::GetProperty, grandma = grandma
grandma = self.grandma
END

;-----
; GetProperty
;-----
PRO mother::GetProperty, mom = mom, _Ref_Extra = Extra
mom = Self.mom
Self->grandmother::GetProperty, _Extra = extra
END

;-----
; GetProperty
;-----
PRO myobj::GetProperty, me=me, _Extra = Extra
me = Self.me
Self->mother::GetProperty, _Extra = extra
END
```

And here is how I changed them to get them to work as you expected them to:

```
;-----
; Getproperty
;-----
PRO grandmother::GetProperty, grandma = grandma, _Ref_Extra=extra
grandma = self.grandma
END
```

```
;-----
; GetProperty
;-----
PRO mother::GetProperty, mom = mom, _Ref_Extra = Extra
mom = Self.mom
Self->grandmother::GetProperty, _Extra = extra
END
```

```
;-----
; GetProperty
;-----
PRO myobj::GetProperty, me=me, _Ref_Extra = Extra
me = Self.me
Self->mother::GetProperty, _Extra = extra
END
```

Here is the result of my changes:

```
IDL> mine = Obj_new('MyObj', 'Mudd', 'Mom', 'Granny')
IDL> mine->SetProperty, Grandma = 'Nana', Mom = 'Mum', Me = 'Pooh'
IDL> mine->GetProperty, Grandma = Grandma, Mom = Mom, Me = Me
IDL> help, grandma, mom, me
GRANDMA      STRING  = 'Nana'
MOM          STRING  = 'Mum'
ME           STRING  = 'Pooh'
```

Cheers,

David

--

David W. Fanning, Ph.D.  
 Fanning Software Consulting  
 Phone: 970-221-0438, E-mail: david@dfanning.com  
 Coyote's Guide to IDL Programming: <http://www.dfanning.com/>  
 Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: keyword inheritance and object inheritance  
 Posted by [btupper](#) on Sun, 24 Mar 2002 21:06:26 GMT

On Sat, 23 Mar 2002 13:45:05 -0700, David Fanning <david@dfanning.com> wrote:

Thanks Ted and David,

Now I get it. And of course, when I reread my references I see that is spelled out quite clearly. I \*really\* did mean to have the MYOBJ pass by reference, but some how I managed to muff it.

Thanks again,

Ben

PS Mom thanks you,too.

PPS Have you ever had a day when, after hanging the laundry on the line to dry, you wonder if you actually washed it? This seems to be mine!

---