
Subject: Re: Important object lesson

Posted by [Mark Hadfield](#) on Tue, 30 Jun 1998 07:00:00 GMT

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

David Fanning wrote in message ...

> Phillip David (pdavid@earthling.net) writes:

>

>> I've been doing some more playing around with objects, and learned that there

>> is a HUGE difference between creating an object with a null constructor and

>> destructor and creating an object with no constructor or destructor.

>> ...

> ...

>

> Actually, I think this is a function of how many directories you have

> on your path. If the INIT and CLEANUP methods are not compiled when

> IDL has to use them, it looks for a file named myclass__init.pro or

> myclass__cleanup.pro in the directories specified by the !PATH

> system variable. If you have a lot of files there, it can take

> a long time!

I'm sure this is the correct explanation.

Interestingly enough, this does not seem to apply to subclasses. If one creates a class

MyClass that inherits from SomeOtherClass, then IDL attempts to resolve calls to

obj_new('MyClass') by first searching for MyClass::Init & if it can't find

that it settles for SomeOtherClass::Init. But as far as I can tell it does

this once only, either at the time MyClass is first defined or (more likely)

the first time it creates a MyClass. After that the rule "when initialising

MyClass call SomeOtherClass::Init" seems to be lodged in its memory banks,

with the result that if you later compile a MyClass::Init it will be ignored.

This explains some confusing experiences I have had with IDL objects.

The reason for this is clear in the light of Phillip's experience. Some IDL objects, eg graphics objects, pick up lots of behaviour from their parents, and if IDL had to resolve every method all the way up the inheritance tree every time it was called, searching the !path at every step, performance would be affected severely.

It also suggests a solution (though not a very elegant one): make all your objects subclasses of something, if only a dummy class, and make sure one of the superclasses has explicit Init and Cleanup methods.

--

Mark Hadfield, m.hadfield@niwa.cri.nz <http://www.niwa.cri.nz/~hadfield/>
National Institute for Water and Atmospheric Research
PO Box 14-901, Wellington, New Zealand
