
Subject: Re: What about real polymorphism ??

Posted by [Michael Wallace](#) on Tue, 14 Dec 2004 18:56:47 GMT

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

> My favorite quote:

>

> Singleton is a social disease. Because Singleton is so easy to
> understand, it is the single pattern that almost anybody who merely
> thumbed through GoF at the bookstore can remember and explain. That
> leads to unwarranted exuberance for the pattern among those who can
> least afford such exuberance.

That's a pretty good quote. Really, you could say that about any of the patterns. One thing that novice programmers will do is discover the Singleton, Decorator, Visitor, Factory or some other pattern and then try to apply the pattern to every possible task usually with great exuberance. What you wind up with are programs that are 10 times as big as what they need to be (if not more) and layer upon layer of needless complexity. I've suffered that same exuberance before, so I know exactly what it feels like and what it feels like to look at the code a year later. :-p

> You *can* implement a singleton in IDL (yes, using a common block or
> system variable). Try a search:

>

> http://groups.google.com/groups?as_q=singleton&as_ugroup=*idl-pvwave*

That's interesting. I had no clue that you could create your own system variables. While not totally ideal, the read only system variable allows you to do what I thought that IDL couldn't do -- be able to enforce the rule of having one and only one class in existence. The previous example just said to have a global variable, which wouldn't be a singleton at all because you could instantiate as many of those variables as you want. I sure didn't know of any way to limit IDL so that you could only create a single instance. I must admit that I am VERY surprised that IDL has this feature and happy to see it. Will wonders never cease?

Mike
