
Subject: Re: "private" and "public" attributes in IDL.
Posted by [Paul van Delst](#) on Thu, 23 Aug 2001 14:21:56 GMT
[View Forum Message](#) <> [Reply to Message](#)

Martin Schultz wrote:

```
>
> JD Smith <jdsmith@astro.cornell.edu> writes:
>
>>>
>>> JD Smith wrote:
>>>>
>> ... and I for one have pined for a native method
>> within IDL to allow this. This does not mean, however, that allowing
>> such carte blanche access is always good idea. Typically, a *small*
>> subset of a class' data fields are useful and stable for public
>> consumption.
>>
>> JD
>
> Wouldn't it be lovely, had the folks at RSI thought about a "public" and "private"
> attribute for object fields?
```

I don't use objects alot (but I'm working on it on the weekends) but your above statement is totally absolutely true. It would've be lovely - even for object-challenged folk like myself. The public and private attribute in fortran 90 (for entire modules, individual variables, structure components and/or internal subprograms) is one of the best additions to that language I reckon.

paulv

p.s. My weekend IDL object tinkerings have certainly improved my Fortran 90 coding techniques - or at the very least how I approach problems in that domain - that's fer sure. And I can write my "get_properties" methods to do whatever I like. :o)

--

Paul van Delst A little learning is a dangerous thing;
CIMSS @ NOAA/NCEP Drink deep, or taste not the Pierian spring;
Ph: (301)763-8000 x7274 There shallow draughts intoxicate the brain,
Fax:(301)763-8545 And drinking largely sobers us again.
 Alexander Pope.

Subject: Re: "private" and "public" attributes in IDL.
Posted by [John-David T. Smith](#) on Thu, 23 Aug 2001 18:00:56 GMT
[View Forum Message](#) <> [Reply to Message](#)

Paul van Delst wrote:

```
>
```

```

> Martin Schultz wrote:
>>
>> JD Smith <jdsmith@astro.cornell.edu> writes:
>>
>>>>
>>>> JD Smith wrote:
>>>> >
>>> ... and I for one have pined for a native method
>>> within IDL to allow this. This does not mean, however, that allowing
>>> such carte blanche access is always good idea. Typically, a *small*
>>> subset of a class' data fields are useful and stable for public
>>> consumption.
>>>
>>> JD
>>
>> Wouldn't it be lovely, had the folks at RSI thought about a "public" and "private"
>> attribute for object fields?
>
> I don't use objects alot (but I'm working on it on the weekends) but your above statement
> is totally absolutely true. It would've be lovely - even for object-challenged folk like
> myself. The public and private attribute in fortran 90 (for entire modules, individual
> variables, structure components and/or internal subprograms) is one of the best additions
> to that language I reckon.
>
> paulv
>
> p.s. My weekend IDL object tinkerings have certainly improved my Fortran 90 coding
> techniques - or at the very least how I approach problems in that domain - that's fer
> sure. And I can write my "get_properties" methods to do whatever I like. :o)

```

We should keep in mind, however, that most "experts" in OOP claim that providing this type of open access is **always** a bad idea, i.e. they lambaste the idea of having anything other than methods public; the RSI engineers may have taken a class or two back in college from a few such experts. Yet, IDL is not a clean and prissy language for academics to build gedanken-algorithms with. It's an warty, meat and potatoes tool for getting work done. Would you be wise to concentrate on flexible accessibility to your object's internals? Absolutely. Should you be forced to, even when a deadline looms? Probably not.

JD
