
Subject: Re: class definition problem

Posted by [J.D. Smith](#) on Tue, 08 Dec 1998 08:00:00 GMT

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

David Fanning wrote:

```
>
> J.D. Smith (jdsmith@astrosun.tn.cornell.edu) writes:
>
>> With a possible addition of
>>
>> tmp={header, field1:0,field2:0,...}
>>
>> before the actual class definition. The point is that the class
>> definition is a perfect place to define other class specific data
>> structures (even if they aren't contained in the class!). I often use
>> the __define procedure to set up special event structures used by the
>> program.
>
> I'm not so sure the class definition is the "perfect" place to
> define data structures used by the program, although I admit
> to sometimes using the technique. In fact, I would
> say this strategy has to be used with a bit of caution.
>
> I foresee a problem in having a number of *different* named
> structures defined with the name HEADER. This could lead to
> messy problems depending upon which programs have been run
> first in the IDL session.
>
> Safer, I think, to put *all* named structures in their
> own structure definition file. Then, of course, you have
> the problem of multiple files with the same name and which
> will be found first in the Path, etc. Perhaps the problem is
> six of one and a half dozen of the other, unless you have
> a specific directory on the path specifically for structure
> definition files. (No one I know would be so organized!)
>
> The secret, I guess, is knowing what you are doing in the
> first place. :-)
```

I suppose I should re-phrase -- it's the perfect place to define data members specific *only* to that class or for use *only* in its methods. This keeps things organized, and provides a convenient place to reference all of your class's associated and auxiliary data (at the bottom of the file). You always know where to go when changes need to be made, and unnecessary re-definitions are avoided (since class__define is called only once).

The standard advice, however, still applies: use named structures only

when you really need to (for replication, event passing, and other special purposes), and always give them unique and specific names. (MAPS_v1_2_HEADER instead of HEADER, for example).

In almost all cases named structures can and should be avoided, but sometimes you just can't do without them. Usually these involve one or more statements such as:

```
foo={Bar_Struct}
```

i.e. the lazy man's structure creation semantic.

JD

--

```
J.D. Smith          [*]   WORK: (607) 255-5842
Cornell University Dept. of Astronomy [*]   (607) 255-6263
304 Space Sciences Bldg.    [*]   FAX: (607) 255-5875
Ithaca, NY 14853          [*]
```
