Subject: Re: Looking for more ideas on code ...
Posted by Craig Markwardt on Wed, 02 Oct 2002 02:44:28 GMT
View Forum Message <> Reply to Message

JD Smith <jdsmith@as.arizona.edu> writes:
> That certainly the canonical "tricky" way to get an array of 1's, and, at
> least on my machine, it's actually faster for most array sizes than:
>
> profile=make_array(n_elements(y),/FLOAT,VALUE=1.)
>
> I started to write this to demonstrate how certain tricks like this can be
> inefficient, only to find it's actually *more* efficient in most cases.
>
> Hmmph. Live and learn.

Interesting performance result! I do it because it allows me to control the type and dimension of the output array pretty simply. The effects of the following statement can be pretty subtle:

```
  y = x*0 + 1.
```

This statement guarantees that Y has the same dimensions as X (except for trailing unit dimensions darnit). But the other nice thing this does is guarantee a certain minimum data type for Y.

Because I am adding the floating point value "1.", Y is guaranteed to be at least floating point. *BUT* if X is double precision, then Y will be double precision as well. This is a nice way to keep the internal precision consistent without resorting to the awkward and error-prone "DOUBLE" keywords that pepper the IDL library.

Craig

--
 -- --------------
Craig B. Markwardt, Ph.D.   EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
 -- --------------