Subject: Trailing array dimensions
Posted by rjp23 on Wed, 01 Jul 2015 09:21:05 GMT
View Forum Message <> Reply to Message

Can someone explain where I'm going wrong with this please?

I have some data which is generally 2 dimensional but occassionally they'll only be 1 data point in the netcdf file, meaning that IDL will initiate it as being 1 dimensional.

For these cases I catch them and reform them to keep the trailing "empty" dimension and put them into a structure.

If I do a help on the structure, this works and keeps the trailing dimension. If however, I do a help on the variable in the structure directly it loses the trailing dimension:

```
IDL> help, data
** Structure <16d2e18>, 12 tags, length=200, data length=197, refs=1:

   my_variable
                  FLOAT     Array[10, 1]

IDL> help, data.my_variable
<Expression>    FLOAT     = Array[10]
```

Any ideas?

Subject: Re: Trailing array dimensions
Posted by rjp23 on Wed, 01 Jul 2015 11:12:06 GMT
View Forum Message <> Reply to Message

On Wednesday, July 1, 2015 at 10:21:07 AM UTC+1, rj...@le.ac.uk wrote:

> Can someone explain where I'm going wrong with this please?

I've got the problem reduced to the fact that I don't know how to test an array to see if it has a trailing dimension.

SIZE reports it only has 1 dimension and ignores the trailing dimension.

Subject: Re: Trailing array dimensions
Posted by Helder Marchetto on Wed, 01 Jul 2015 11:58:59 GMT
View Forum Message <> Reply to Message

On Wednesday, July 1, 2015 at 1:12:09 PM UTC+2, rj...@le.ac.uk wrote:

> On Wednesday, July 1, 2015 at 10:21:07 AM UTC+1, rj...@le.ac.uk wrote:

>> Can someone explain where I'm going wrong with this please?
>
>
> I've got the problem reduced to the fact that I don't know how to test an array to see if it has a trailing dimension.
>
> SIZE reports it only has 1 dimension and ignores the trailing dimension.

Hi,
how about using reform?

```
IDL> my_variable=indgen(10,1)
IDL> if size(my_variable,/n_dimensions) eq 1 then data = {my_variable:reform(my_variable,10,1)}
else data = {my_variable:my_variable}
IDL> help, data
** Structure <18c8f8d0>, 1 tags, length=20, data length=20, refs=1:
   MY_VARIABLE     INT       Array[10, 1]
```

Cheers,
Helder

Subject: Re: Trailing array dimensions
Posted by Craig Markwardt on Wed, 01 Jul 2015 23:02:30 GMT

View Forum Message <> Reply to Message

On Wednesday, July 1, 2015 at 2:21:07 AM UTC-7, rj...@le.ac.uk wrote:

```
>  Can someone explain where I'm going wrong with this please?
>
>  I have some data which is generally 2 dimensional but occassionally they'll only be 1 data point in the netcdf file, meaning that IDL will initiate it as being 1 dimensional.
>
>  For these cases I catch them and reform them to keep the trailing "empty" dimension and put them into a structure.
>
>  If I do a help on the structure, this works and keeps the trailing dimension. If however, I do a help on the variable in the structure directly it loses the trailing dimension:
>
>  IDL> help, data
>  ** Structure <16d2e18>, 12 tags, length=200, data length=197, refs=1:
>
>     my_variable
>                    FLOAT     Array[10, 1]
>
>  IDL> help, data.my_variable
>  <Expression>    FLOAT     = Array[10]
>
>  Any ideas?
```

I've never gotten this to work right. I complained loudly about this in the year 2000, fifteen years ago, and nothing has changed.

The answer is, if you are using 2D arrays or larger in structures, prepare for array jiggery-pokery to happen behind your back. The only thing you can do is REFORM() it when you use it.

Craig