Subject: Dropped dimensions?
Posted by Craig Markwardt on Sun, 28 Mar 1999 08:00:00 GMT
View Forum Message <> Reply to Message

I really enjoy programming in IDL. Because of dynamic typing and dimensioning of variables, and the inherent vector nature of most operators, the language itself can be exceptionally powerful. I find myself doing some very mind-twisting things with ease in IDL, which become very difficult if I have to translate to C.

However, sometimes IDL simply drives me up the wall. No surprise, it's a problem with IDL silently dropping the last dimension of a vector variable. Recently we've seen it causing havoc with matrix multiplication. Here's another example:

Goal: use TOTAL to total one dimension in an array, A. A is three dimensional, but can have any dimensions (ie, it can even be 1x1x1). Typically I want to total the last dimension of the three.

ATTEMPT 1: **

```
A = DBLARR(N1, N2, N3)
... processing ...
TOT = TOTAL(A, 2)
```

ANALYSIS: Looks good right? Wrong, because IDL can silently drop any number of trailing dimensions of size 1 from the array, so occasionally the array doesn't have a third dimension to total. Okay, we can REFORM it and try again.

ATTEMPT 2: **

```
A = DBLARR(N1, N2, N3)
... processing ...
A = REFORM(A, N1, N2, N3, /OVERWRITE)  ; Make sure dimensions are correct
TOT = TOTAL(A, 2)
```

ANALYSIS: In fact, you will see this formalism a lot in my code. I usually reform an array instinctively after I create it, just to be sure it has the dimensions I ask for! Okay but this still has a problem because sometimes, if A is a 1x1x1 array at the start, the processing can leave only a scalar. Surprise again! Because REFORM() does not accept scalars. So this is what I am left with:

ATTEMPT 3: **

```
A = DBLARR(N1, N2, N3)
... processing ...
IF N_ELEMENTS(A) EQ 1 THEN A = [A]     ; Make sure it's an array
A = REFORM(A, N1, N2, N3, /OVERWRITE)  ; Make sure dimensions are correct
TOT = TOTAL(A, 2)
```

ANALYSIS: Okay, this works in most cases. But it's a lot of hoops to jump through for a simple operation.

What is the moral of the story?

For IDL programmers: you have to be very careful about where your array variables get silently REFORMed. REFORM them yourself at critical points.

For RSI:
 * REFORM should operate on scalars too.
 * TOTAL should ignore missing final dimensions, since those dimensions can be dropped.
 * Dimensions should not be dropped! I do appreciate when that happens sometimes, but it usually happens at random and dangerous moments. I would like to have explicit control over when it happens. Something like a RELAX procedure which "relaxes" unneeded final dimensions.

Thanks for staying with me on this tirade. Back to shiny happy thoughts now.

Craig

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

Subject: Re: Dropped dimensions?
Posted by MarioIncandenza on Thu, 28 Jun 2012 19:16:47 GMT
View Forum Message <> Reply to Message

On Sunday, March 28, 1999 1:00:00 AM UTC-7, Craig Markwardt wrote:

> For RSI:
> * REFORM should operate on scalars too.
> * TOTAL should ignore missing final dimensions, since those dimensions
> can be dropped.
> * Dimensions should not be dropped! I do appreciate when that happens
> sometimes, but it usually happens at random and dangerous moments.
> I would like to have explicit control over when it happens. Something
> like a RELAX procedure which "relaxes" unneeded final dimensions.

Has the behavior of IDL w/r/t silently REFORMing away dimensions changed in the last 13 years?

Have any clever solutions emerged where I do not need to use an IF(N_DIM eq 1) for operations where N_DIM might sometimes be 1? Just checking.

--Edward H.

Subject: Re: Dropped dimensions?
Posted by wlandsman on Thu, 28 Jun 2012 19:45:45 GMT
View Forum Message <> Reply to Message

On Sunday, March 28, 1999 3:00:00 AM UTC-5, Craig Markwardt wrote:
Okay but this still has a
> problem because sometimes, if A is a 1x1x1 array at the start, the
> processing can leave only a scalar. Surprise again! Because REFORM()
> does not accept scalars. So this is what I am left with:

How do you end up with a scalar? At least since IDL V7.1, I find,

```
IDL> a = dblarr(1,1,1)
IDL> help,a
A               DOUBLE    = Array[1]
IDL> b = reform(a,1,1,1)
IDL> help,b
B               DOUBLE    = Array[1, 1, 1]
```

Subject: Re: Dropped dimensions?
Posted by wlandsman on Thu, 28 Jun 2012 20:07:39 GMT
View Forum Message <> Reply to Message

On Thursday, June 28, 2012 3:45:45 PM UTC-4, wlandsman wrote:
> On Sunday, March 28, 1999 3:00:00 AM UTC-5, Craig Markwardt wrote:

>
> How do you end up with a scalar? At least since IDL V7.1, I find,

To answer my own question, the subsequent processing might convert a 1 element array to a scalar. I do try to have my own programs avoid such conversions, but it can be hard to maintain consistency. --Wayne

>
> IDL> a = dblarr(1,1,1)
> IDL> help,a
> A   DOUBLE   = Array[1]
> IDL> b = reform(a,1,1,1)
> IDL> help,b
> B   DOUBLE   = Array[1, 1, 1]

Subject: Re: Dropped dimensions?
Posted by Craig Markwardt on Sun, 01 Jul 2012 05:51:23 GMT
View Forum Message <> Reply to Message

On Thursday, June 28, 2012 3:07:39 PM UTC-5, wlandsman wrote:
> On Thursday, June 28, 2012 3:45:45 PM UTC-4, wlandsman wrote:
>> On Sunday, March 28, 1999 3:00:00 AM UTC-5, Craig Markwardt wrote:
>
>>
>> How do you end up with a scalar? At least since IDL V7.1, I find,
>
> To answer my own question, the subsequent processing might convert a 1 element array to a scalar. I do try to have my own programs avoid such conversions, but it can be hard to maintain consistency. --Wayne

I try too, but sometimes it just works out that a scalar pops out.

Wow, this thread started 13 years ago!

Craig

Subject: Re: Dropped dimensions?
Posted by lecacheux.alain on Sun, 01 Jul 2012 07:43:26 GMT
View Forum Message <> Reply to Message

On 1 juil, 07:51, Craig Markwardt <craig.markwa...@gmail.com> wrote:
> On Thursday, June 28, 2012 3:07:39 PM UTC-5, wlandsman wrote:
>> On Thursday, June 28, 2012 3:45:45 PM UTC-4, wlandsman wrote:
>>> On Sunday, March 28, 1999 3:00:00 AM UTC-5, Craig Markwardt wrote:
>
>>> How do you end up with a scalar? At least since IDL V7.1, I find,
>
>> To answer my own question, the subsequent processing might convert a 1 element array to a scalar. I do try to have my own programs avoid such conversions, but it can be hard to maintain consistency. --Wayne
>
> I try too, but sometimes it just works out that a scalar pops out.
>
> Wow, this thread started 13 years ago!
>
> Craig
>
>

I do not feel that the implicit reforming of arrays dimensions is a
too strong problem (even if discussed for 13 years!). First, because
only the last dimension of an array can be concerned (with possible

recursion); second, because the problem arises only in the case of a statement explicitly asking for dimension (like total, max, mean...). Then if you like to be generic, you must use, as Craig's said, some construct like total(reform([array], mydims), dim). In all other cases, this IDL rule is an advantage rather than an inconvenience. alain.
