Subject: Why does IDL strip unary dimensions from structure elements?
Posted by Thomas A McGlynn on Thu, 26 Oct 1995 07:00:00 GMT
View Forum Message <> Reply to Message

I've been writing some IDL code which does a lot of generic
stuff with dyamically defined structures, and I've run into
a bug/feature that is causing some problems.

Suppose you have an array of structures, s, with an element
e which is itself an array, e.g,

s = replicate({e:intarr(e1,e2,e3), s1)

If I now extract e from the structure, e.g.,

var = s.(0)

I'll get var with dimensions, e1,e2,e3,s1. So far so good.
However, if the last dimension, or the last several dimensions
of this list are 1, then these dimensions will be stripped
so that the dimensionality of var will be less than expected.
E.g.,

s = replicate({e:intarr(10,1,1)},1)
b = s.(0)

gives b which is a one dimensional vector, not four dimensions.

IDL is consistent: print, size(s.(0)) yields
 1 10 2 10

Since it's also an error to have more dimensions on the right
side of an equation than on the left, this means that I have
to keep track of the fact that IDL silently drops these dimensions.
I am also a bit concerned if this behavior may change in different
versions of IDL since my code is intended to be very portable.

Any ideas on clever ways to deal with this? I've seen this with
IDL 3.6 and 4.0.1 on OSF and Ultrix. Is this behavior standard?

Thanks for your help,
Tom McGlynn
mcglynn@grossc.gsfc.nasa.gov

Subject: Re: Why does IDL strip unary dimensions from structure elements?
Posted by thompson on Thu, 26 Oct 1995 07:00:00 GMT

Thomas A McGlynn <mcglynn@grossc.gsfc.nasa.gov> writes:

> I've been writing some IDL code which does a lot of generic
> stuff with dyamically defined structures, and I've run into
> a bug/feature that is causing some problems.

Stuff deleted, about the fact that IDL strips off trailing unary dimensions.

We've also run into this "feature", and it's a big pain. I'm sure it's there because somebody wanted it somewhere along the line, or thought it was a good idea.

It appears in many places, not only when working with structures. For example

```
IDL> a = indgen(3,1,1) & help,a
A               INT       = Array(3)
```

or

```
IDL> a = reform(a,3,1,1) & help,a
A               INT       = Array(3, 1, 1)
IDL> a = float(a) & help,a
A               FLOAT     = Array(3)
```

or

```
IDL> a = reform(a,3,1,1) & help,a
A               FLOAT     = Array(3, 1, 1)
IDL> a = a /3 & help,a
A               FLOAT     = Array(3)
```

The only way I've come across to get around this problem is to use the SIZE function to get the original dimensions, and then REFORM to put them back, e.g.

```
SZ = SIZE(A)
A = A / 3
A = REFORM(A, SZ(1:SZ(0)), /OVERWRITE)
```

As I said, it's a real pain.

Bill Thompson