Subject: Trailing singular dimensions disappearing. Call for discussion.
Posted by steinhh on Tue, 28 Feb 1995 10:00:02 GMT
View Forum Message <> Reply to Message

I have come across the following annoying "feature" of IDL through
a programming project: If you type e.g.,

```
IDL> HELP, fltarr(10,1,1)  ; you get:
<Expression>    FLOAT     = Array(10)
```

But if you say

```
IDL> HELP, fltarr(1,1,10)  ; you get:
<Expression>    FLOAT     = Array(1, 1, 10)
```

In our applications, it matters that the data arrays retain the
correct number of dimensions, as we have e.g., axis names for each
dimension. The above behaviour is clearly inconsistent, and IDL
is definitely doing something that is not intuitively obvious,
nor is it desirable at all times.

The behaviour I would like is:

Whenever I create an array with so and so many dimensions
(singular as well..), I'd like it to stay that way, until explicitly
told otherwise.

When I'm indexing an array with a scalar, I'd like the dimension to
disappear ("reformed away"), e.g.:

```
IDL> a=fltarr(10,10)
IDL> help,a(1,1)
<Expression>    FLOAT     =       0.00000
```

This is OK, and it's what happens with today's version, but if I do
the following:

```
IDL> help,a(1,*)
<Expression>    FLOAT     = Array(1, 10)
```

or, to make the example complete, we try:

```
IDL> help,a(*,1)
<Expression>    FLOAT     = Array(10)
```

Here I would have liked both expressions (a(1,*) and a(*,1)) to
return the same thing (an Array(10)). I'd still like to be able to
retain a singular dimension by indexing with an array, e.g.:

```
IDL> help,a(*,[1])
<Expression>    FLOAT     = Array(10, 1)
```

So, to sum up: Arrays stay unaltered until otherwise is specified. Dimensions disappear when indexed with a scalar value, but remain when indexed with an array (even a single-element one).

In order to make these changes go through with RSI, I'd like to hear (in this newsgroup) from as many as possible. Of course, if anyone has large sets of procedures that are tuned directly to today's behaviour, they are welcome to raise their voice, but I would argue that their programs would be better off rewritten with a logical behaviour instead of sneaking dimensions out the back door.

Sincerely,
Stein Vidar H. Haugan

Subject: "Tweening" or Morphing with IDL
Posted by lmudge on Mon, 20 Mar 1995 06:28:10 GMT

View Forum Message <> Reply to Message

Does anyone have any experience with a process known as "tweening" in IDL. This process, closely related to morphing involves interpolating between two vector images (ie polygons) producing the coordinates of the intermediate images (polygons). This is used when producing a series of images to animate as it produces much smoother transitions between images.

I have looked at the morphing demo that comes with IDL v3.6.1 and this gives some ideas of how it may be possible too go about this.

End Of Message
