Subject: Re: Addressing 3D arrays different from 2D arrays?
Posted by Spon on Tue, 06 Nov 2007 17:30:05 GMT
View Forum Message <> Reply to Message

On Nov 6, 5:13 pm, Jaron Kurk <jaron.k...@googlemail.com> wrote:
> Dear readers,
>
> Apologies if this question has long been answered, but I could not
> find anything on it.
>
> Is there some fundamental difference in addressing 3D arrays and 2D
> arrays? In IDL 6.3 (and GDL), the following code fills a 2D array with
> a circle of 1's but a slice of a 3D array with a square of 1's, while
> I would expect just the same area filled with 1's as for the 2D case.
> Note that the use of reform() does not cause the difference, I have
> checked that.
>
> xidx=[5,4,5,6,3,4,5,6,7,4,5,6,5]
> yidx=[3,4,4,4,5,5,5,5,5,6,6,6,7]
> test2d = bytarr(10,10)
> test3d = bytarr(10,10,10)
> test2d[xidx,yidx] = 1
> test3d[0,xidx,yidx] = 1
> print,test2d,total(test2d)
> print,reform(test3d[0,*,*]),total(test3d)
>
> If anybody could enlighten me, I would appreciate it!
>
> Jaron Kurk.

I can get rid of it, but I'm not sure why you're getting a square (as
opposed to just junk):

```
xidx = [5,4,5,6,3,4,5,6,7,4,5,6,5]
yidx = [3,4,4,4,5,5,5,5,5,6,6,6,7]
zidx = REPLICATE (0, N_ELEMENTS (xidx))
test2d = bytarr(10,10)
test3d = bytarr(10,10,10)
test2d[xidx,yidx] = 1
test3d[zidx,xidx,yidx] = 1
print,test2d,total(test2d)
print,reform(test3d[0,*,*]),total(test3d)
```

Your test3d array wasn't shifting your first two arrays by a whole
dimension, just one element.

Chris