Subject: Re: adding a 4th dimension to 3D array during concatenation
Posted by wlandsman on Thu, 17 Mar 2016 03:46:16 GMT
View Forum Message <> Reply to Message

On Tuesday, March 15, 2016 at 6:37:18 PM UTC-4, Wayana Dolan wrote:
You can go ahead with the list suggestions but if you do know how many arrays you want to concatenate, then it is even more efficient to build your array beforehand.

```
outarr = fltarr(91,41,33,16)
for i=0,15 do begin
   ....   Make an array x with dimensions (91,41,33)
   outarr[0,0,0,i] = x
endfor

x = transpose(x, [3,0,1,2])
```

Now for efficiency reasons it is better to make the last dimension 16 rather than the first. But if you really want the first dimension to be the "counting" dimension then you can use TRANSPOSE() as above to rearrange things.

> So outside a loop I start out with an empty array (ex. array=[ ]). Then each time through the loop, I make an array with 3 dimensions (for example, array x has dimensions[91, 41, 33]), and then concatinate it to the previous array. (ex. array=[array, x]).
>
> Lets say we run through the loop 16 times. What I'd like as a result is something that has dimensions like this [16, 91, 41, 33].
>
> I'm not sure how to do this... I've looked at IDL coyote's concatenation tutorial, and still am having trouble.
>
> I'm pretty new to coding period, so this is a challenge. Any ideas?