Subject: Implementing a DIMENSION keyword
Posted by Matthew Argall on Thu, 11 Apr 2013 22:30:05 GMT
View Forum Message <> Reply to Message

I would like to write a wrapper function for the FFT routine and implement a sliding window. I want to keep the functionality of the DIMENSION keyword to FFT, but am not sure of the best way. I know about Craig Markwardt's CMAPPLY, but was going for something a little more straight-forward.

What I am doing now is transposing the array so that the desired dimension is the leading dimension, then applying the FFT as in the following pseudo-code

```
array = put_dimension_first(array, dimension)
result = fft(array[start_index:end_index, *, *, *, *, *, *], DIMENSION=1)
result = put_dimension_first(result, dimension, /UNDO)
```

This works because IDL ignores the extra "*"s for dimensions that do not exist. Not sure if this is a good idea or not...

Subject: Re: Implementing a DIMENSION keyword
Posted by Matthew Argall on Thu, 11 Apr 2013 22:34:34 GMT
View Forum Message <> Reply to Message

There should be an extra * in there... the maximum number of dimensions in IDL is 8, so this would work in general...