Subject: Re: forcing variable definition in IDL?
Posted by William Daffer on Tue, 01 May 2001 20:26:29 GMT
View Forum Message <> Reply to Message

"Liam E. Gumley" <Liam.Gumley@ssec.wisc.edu> writes:

> William Daffer wrote:
>>
>> davidf@dfanning.com (David Fanning) writes:
>> [...]
>>
>>> Don't bother. IDL scalars *are* single element arrays:
>>>
>>> IDL> a=5
>>> IDL> a[0] = 6 & Print, a
>>>
>>
>> Um... Not true.
>>
>> IDL> a=['foo|bar']
>> IDL> print,strsplit(a,'|',/extract)
>> % STRTOK: Expression must be a scalar in this context: STRING.
>> % Execution halted at: STRSPLIT 24
>> /usr/local/rsi/idl_5.3/lib/strsplit.pro
>> % $MAIN$
>> IDL> retall
>> IDL> print,strsplit(a[0],'|',/extract)
>> foo bar
>> IDL>
>>
>> There are some other RSI supplied code where one sees this behavior.
>>
>> By the way, this is idl 5.3. I haven't checked idl 5.4.
>
> An array with one element is an *array*, i.e., it has one dimension:
>
> IDL> a = [25]
> IDL> help, a
> A INT = Array[1]
> IDL> print, size(a, /n_dimensions)
> 1
>
> A single subscripted array element is a *scalar expression*, i.e., it
> has no dimensions:
>
> IDL> a = [1, 2, 3, 4, 5]
> IDL> help, a[0]
> <Expression> INT = 1

> IDL> print, size(a[0], /n_dimensions)
> 0
>
> A scalar may be treated as though it were a single subscripted array
> element. However, as shown above, a scalar expression has no dimensions:
>
> IDL> a = 100
> IDL> help, a
> A INT = 100
> IDL> help, a[0]
> <Expression> INT = 100
> IDL> print, size(a, /n_dimensions)
> 0
> IDL> print, size(a[0], /n_dimensions)
> 0
>
> The implementer of STRTOK (which is called by STRSPLIT) is therefore
> checking for an input argument which has no dimensions.
>
> Cheers,
> Liam.
> http://cimss.ssec.wisc.edu/~gumley/

Um... so you're agreeing with me when I say that David's remark is untrue?

whd
--
Outside of a dog a book is man's best friend.
Inside of a dog it's too dark to read
Groucho Marx