Subject: Re: Error: Array has a corrupted descriptor
Posted by Karl Schultz on Tue, 28 Nov 2006 17:12:57 GMT
View Forum Message <> Reply to Message

On Tue, 28 Nov 2006 09:06:11 -0700, David Fanning wrote:

> Gongqin Shen writes:
>
>> Thank you for paying attention to my question. Bascially, "do
>> something here" is just some basic array manipulation like:
>> arr = arr3D[1:2, *]
>> arrInd[0, *] -= a
>> arrInd[1, *] -= b
>> arr[arrInd] = 1B
>> aLabeled = Label_Region(arr)
>>
>> FOR j = 1, MAX(aLabeled) DO BEGIN
>> ithIdx = (WHERE(aLabeled EQ j))[0]
>> outputList = [outputList, ithIdx]
>> ENDFOR
>>
>> I hope your eagle eye can find out where the glitch comes from. :-).
>
> I think the likely suspect is this line:
>
> outputList = [outputList, ithIdx]
>
> At the *very* least, I would write this line like this:
>
> outputList = [Temporary(outputList), ithIdx]
>
> But I think you might be better off making output list as big
> as you need it (or bigger) and then filling it. I suspect
> the constant recreating of this array is fragmenting memory
> like crazy and resulting in your problems.
>
> void = Where(aLabeled GT 0, count)
> outputList = Lindgen(count)
>
> Cheers,
>
> David

The corrupted array descriptor error message is almost always caused by a programming error in an IDL system routine or user-written DLM function. You generally cannot cause this error by just moving array elements around since IDL performs bounds checking. This code is also pretty mature and we would have fixed any problems by now. Likewise, a storage

fragmentation issue that leads the inability to allocate a large block of storage would result in a message about storage allocation, not a corrupted array descriptor.

Unless there is a user-written DLM in the mix, I suspect a bug in LABEL_REGION. It would be really good if Gongqin could create a reproduce case and get it to ITTVIS Tech Support.

Karl
