Subject: Re: openr and /get_lun
Posted by John-David T. Smith on Mon, 17 Apr 2000 07:00:00 GMT
View Forum Message <> Reply to Message

"Robert S. Mallozzi" wrote:
>
> In article <onitxkz7p7.fsf@cow.physics.wisc.edu>,
>   Craig Markwardt <craigmnet@cow.physics.wisc.edu> writes:
>>
>> I have noticed that the use of /GET_LUN and ERROR keywords to openr is
>> not as helpful as I would have hoped. Do other have this experience?
>> The problem is that when an error occurs, it is hard to know whether
>> the file unit was "gotten" or not.
>>
>> For example:
>>
>> pro test1
>>   openr, unit, filename, /get_lun, error=err
>>   free_lun, unit
>> end
>>
>> If there was an error, then it is possible that UNIT was never set,
>> and is hence undefined. FREE_LUN doesn't take undefined variables.
>
> I guess I never thought about it too much, because if
> there is an error with the OPEN, then I should handle
> it somehow:
>
>   OPENR, fl, 'nofile', /GET_LUN, ERROR = error
>   IF (error NE 0) THEN BEGIN
>   MESSAGE, /CONTINUE, 'Could not open file.'
>   RETURN
>   ENDIF
>   .
>   .
>   .
>   FREE_LUN, fl
>
> Otherwise, if you don't want to handle the error, you can just
> free the unit number conditionally, as I am sure you know:
>
>   IF (error EQ 0) THEN FREE_LUN, fl
>
>
> I sure wish we had a boolean datatype - the mistake of
> using something like "IF (NOT error) THEN" is one that
> is really a pain to find, although it certainly makes
> your code much more readable.

We don't need a boolean data type... we need IF to examine not just the first bit of the value, but the whole thing, and use C's 0=false, anything else =true paradigm. Here's hoping.

if NOT 2 then print,"this isn't right!"

JD

--
J.D. Smith |*| WORK: (607) 255-5842
Cornell University Dept. of Astronomy |*| (607) 255-6263
304 Space Sciences Bldg. |*| FAX: (607) 255-5875
Ithaca, NY 14853 |*|