
Subject: Re: openr and /get_lun
Posted by [davidf](#) on Fri, 14 Apr 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

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.
>
> If there is error checking to do, I don't know exactly what it should
> be. So I find myself explicitly doing this instead:
>
> pro test2
> get_lun, unit
> openr, unit, filename, error=err
> free_lun, unit
> end
>
> Comments?

I guess I'd use the ol' IF N_Elements(lun) NE 0 THEN Free_Lun, lun

Cheers,

David

--

David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: openr and /get_lun

Posted by [majewski](#) on Sun, 16 Apr 2000 07:00:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

you could always jump over the access of the file with a GOTO...

```
pro test_lun_free
```

```
    filename = 'c:\dne.dat'
```

```
    openr, unit, filename, /get_lun, error=err
```

```
        IF (err NE 0) then goto, bad_file
```

```
        ;Insert your read statement here
```

```
        ;Insert your read statement here
```

```
        print, 'the file exists - lun#', strcompress(string(unit),  
/remove_all), ' set'
```

```
        free_lun, unit
```

```
        ;Insert the rest of your program here
```

```
bad_file: If err ne 0 then print, 'lun not set'
```

```
end
```

- i think this is what you want

leon

On 14 Apr 2000 16:36:04 -0500, Craig Markwardt
<craigmnet@cow.physics.wisc.edu> wrote:

>

> 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.

>

> If there is error checking to do, I don't know exactly what it should
> be. So I find myself explicitly doing this instead:

>

```
> pro test2
>  get_lun, unit
>  openr, unit, filename, error=err
>  free_lun, unit
> end
>
> Comments?
>
> Craig
>
> --
> -----
> Craig B. Markwardt, Ph.D.      EMAIL:  craigmnet@cow.physics.wisc.edu
> Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
> -----
```

Leon Majewski

Remote Sensing & Satellite Research Group
Curtin University of Technology, Perth, Australia

email: majewski@ses.curtin.edu.au

Subject: Re: openr and /get_lun
Posted by [mallors](#) on Sun, 16 Apr 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

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.

-bob

--

```
~~~~~
Robert S. Mallozzi                256-544-0887
                                Mail Code SD 50
http://gammaray.msfc.nasa.gov/    Marshall Space Flight Center
http://cspar.uah.edu/~mallozzir/  Huntsville, AL 35812
~~~~~
```

Subject: Re: openr and /get_lun
Posted by [Nando lavarone](#) on Mon, 17 Apr 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

<!doctype html public "-//w3c//dtd html 4.0 transitional//en">
<html>

Craig Markwardt wrote:

<blockquote TYPE=CITE>I have noticed that the use of /GET_LUN and ERROR

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.

<p>pro test1

```
<br>&nbsp; openr, unit, filename, /get_lun, error=err
<br>&nbsp; free_lun, unit
<br>end
```

<p>pro test2

```
<br>&nbsp; get_lun, unit
<br>&nbsp; openr, unit, filename, error=err
<br>&nbsp; free_lun, unit
<br>end
```


you have to trap the error.

it is the open that can fail, so you have to check the err variable:

<p>pro test2

[illegible]

```
You can also use the more flexible error trapping CATCH procedure.</pre>
```

```
It is general and not only for file management:
```

```
<pre></pre>
```

```
pro testOpen<br>
```

[illegible]

