
Subject: Re: Warning: IDL 8.0 alters the behaviour of existing valid programs without any notice!

Posted by [svhhaugan](#) on Wed, 18 Aug 2010 13:30:55 GMT

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

On Aug 18, 2:14 pm, Maarten <maarten.sn...@knmi.nl> wrote:

> On Aug 18, 1:59 pm, svhhaugan <s.v.h.hau...@gmail.com> wrote:

>

>> pro test,data

>> catch,error

>> if error ne 0 then begin

>> catch,/cancel

>> print,"Guess that didn't work"

>> return

>> end

>> data[where(data eq min(data)-1)] = 50

>> end

>

>> Perfectly valid code, makes sense in IDL 7.1 although the catch,error

>> part would most likely be done "with human intervention".

>

> I just looked up the help for where in IDL 7. This is what is

> prominently displayed on that page:

>

> Note: When WHERE Returns -1

>

> If all the elements of Array_Expression are zero, WHERE returns a

> scalar integer with a value of -1. Attempting to use this result as an

> index into another array results in a "subscripts out of bounds"

> error. In situations where this is possible, code similar to the

> following can be used to avoid errors:

>

> ; Use Count to get the number of nonzero elements:

> index = WHERE(array, count)

>

> ; Only subscript the array if it's safe:

> IF count NE 0 THEN result = array[index]

>

> And yes, you can use the try/catch exception method of handling this,

> but I would pose that testing for an item not found message and then

> using an IF statement to handle that common situation makes more sense

> than relying on an error. So in that sense the code was never valid or

> sensible in the first place.

As written, it is just for "demonstrational purposes". But I bet a lot of code out there is written to *break* when tested (interactively) with parameters that just won't work. This code will no longer break, it will silently do something different

from what was originally intended.

It was a documented feature ("prominently displayed") of IDL 7 that such usage would cause a crash or trigger a catch statement ("Attempting to use this result as an index ... results in [an] error").

For IDL 8, the rules have suddenly been changed, with no warning and no way to turn it off (other than staying with version 7).

> Does that mean no one will be caught off-guard by this? No. Is that an
> issue? Some people just get what they deserve. The added benefit of
> being able to count from the end of an array is certainly worth some
> trouble.

I agree that counting from the end of the array is a Good Thing. But I think having to insert a "compile_opt strictnegarrsubs" (or something of that sort) in every existing routine written for IDL 7 would be enough trouble. Right now, not even that is possible.

And although it's fine to say "some people just get what they deserve", the problem is that these people won't necessarily notice that anything is wrong, and others may rely on the false results.
