Subject: Re: IDL 8.0 compile_opt changes
Posted by chris_torrence@NOSPAM on Mon, 21 Dec 2009 17:23:12 GMT
View Forum Message <> Reply to Message

Just a quick note about the negative indices. They are *only* allowed in ranges or as a scalar index. They are *not* allowed in arrays of indices.

For example:

a[0:-1] In IDL80, returns all the elements

a[-1] In IDL80, returns the last element

a[[-1,0,1,2]] will either truncate the -1 to 0, or will throw an error if the "strictarrsubs" compile option is set. This behavior will remain the same in IDL80. We will not change it.

So, for example, David's example:

```
z= where(bad ne 0)
if z[0] ne -1 then begin
  lots of stuff
endif
```

This will still work *fine* in IDL80.

Now, in IDL80 the following might cause a problem:

```
 z= where(bad ne 0)
 do something with bad[z] (without checking for -1)
```

In this case, if "where" returns a scalar -1, then bad[z] will indeed return the last element. However, it would be bad to use the where result without checking for a -1...

So, in summary, in IDL 8.0:
1. Negative indices are only used for ranges or a scalar index
2. Negative indices within an index array will still behave the same (either clip or throw an error)
3. The behavior of Where remains unchanged (and you still need to check for a -1 result, as always)

Returning to the compile_opt discussion...

-Chris
ITTVIS