
Subject: Re: Pass-By-Reference question.

Posted by [Mark Hadfield](#) on Wed, 10 Mar 2004 23:55:17 GMT

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

Craig Markwardt wrote:

> Mark Hadfield <m.hadfield@niwa.co.nz> writes:

>

>> Not so. As Tim said, keywords passed in via inheritance override

>> keywords supplied explicitly. This is a very useful feature and

>> virtually all of my code relies on it.

>>

>

>

> Mark, are you relying on a documented feature of IDL, or are you just

> assuming that because it works, it's right? I've never assumed that

> keyword inheritance works by overriding explicit values. In fact, I

> would have assumed the exact opposite, namely that the explicit

> overrides the generic.

This behaviour *is* documented for value inheritance, eg under the topic "Keyword Inheritance", heading "Writing a wrapper routine", sub-heading "By Value" the IDL 6.0 help file says the following"

Note that keywords passed into a routine via `_EXTRA` override previous settings of that keyword. For example, the call:

```
PLOT, a, b, COLOR = color, _EXTRA = {COLOR: 12}
```

specifies a color index of 12 to PLOT.

Under the sub-heading "By Reference" it says something very similar, but I must admit I don't understand why structures get introduced here:

These inherited keywords are then passed from TEST to the PLOT routine using the `_EXTRA` keyword. Note that keywords passed into a routine via `_EXTRA` override previous settings of that keyword. For example, the call:

```
PLOT, a, b, COLOR = color, _EXTRA = {COLOR: 12}
```

specifies a color index of 12 to PLOT. Also note that we are passing a structure (the by value format used by `_EXTRA`) as the value of the extra keyword to a routine that uses the by reference keyword inheritance mechanism (`_REF_EXTRA`). There is no problem in doing this, because each routine establishes its own inheritance mechanism independent of any other routines that may be calling it. However, any keyword values that are changed within PLOT will fail to be returned to the caller due to the use of the by-value mechanism.

Anyway, back in Aug 2000, during the beta test period of IDL 5.4, there was a long thread on this newsgroup with the title "Keyword precedence", involving me & JD mainly, during which I noted that there was an inconsistency between the keyword precedence rules for value and reference inheritance, and I argued that they should both behave the way described above (inherited keywords override explicit keywords). I also raised this with RSI and persuaded them to change the behaviour for reference inheritance by the time IDL 5.4 final was released.

If you review that thread (though I doubt you'll have the stamina to read the whole thing) you may note that I began by saying that I had until recently assumed the contrary (explicit should override inherited). My first message in that thread explains why I changed my mind: it allows one to write cleaner wrapper routines.

--

Mark Hadfield "Ka puwaha te tai nei, Hoea tatou"
m.hadfield@niwa.co.nz
National Institute for Water and Atmospheric Research (NIWA)
