
Subject: Re: Pointer Problem

Posted by [davidf](#) on Tue, 15 Jul 1997 07:00:00 GMT

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

J.D. Smith writes:

```
> This is a possible pointer problem which may affect some of you.
>
> IDL> b=ptr_new(notavalidvariable)
> IDL> help,b
> B          POINTER  = <PtrHeapVar2>
> IDL> print,ptr_valid(b)
>    1
>
> Note here that 'notavalidvariable' is a fake, undefined variable. B is
> assigned to an undefined heap variable, as if /ALLOCATE_HEAP had been
> specified. Consequently, ptr_valid see's b as valid. This behavior is
> mildly irksome in that the convenience of assigning pointers without
> checking the contents of that which they are being assigned to is lost.
> It is also non-intuitive.
```

While this behavior may be irksome, it does have the advantage of being consistent with how variables are treated in IDL. "Undefined" has always been a perfectly valid variable type. If you think of a "valid" pointer as being one that points to memory where data can be stored, then this behavior makes quite a lot of sense.

If you just want to know if your variable is "undefined", then you can write your own IS_DEFINED function. Here is mine. It works with pointers as well as variables. Note in the code below that the variable b is defined as a valid pointer type variable, but that the thing it points to is undefined.

```
FUNCTION IS_DEFINED, variable
s = SIZE(variable)
IF s(s(0)+1) EQ 0 THEN RETURN, 0 ELSE RETURN, 1
END
```

```
IDL> Print, IS_DEFINED(notavalidvariable)
0
IDL> b = Ptr_New(notavalidvariable)
IDL> Print, IS_DEFINED(b)
1
IDL> Print, IS_DEFINED(*b)
0
```

Cheers,

David

----- --
David Fanning, Ph.D.
Fanning Software Consulting
Customizable IDL Programming Courses
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: <http://www.dfanning.com>
IDL 5 Reports: <http://www.dfanning.com/documents/anomaly5.html>
