
Subject: Re: Tough System Variable Question for the IDL Gurus
Posted by [Foldy Lajos](#) on Wed, 11 Nov 2009 22:26:47 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Wed, 11 Nov 2009, David Fanning wrote:

```
> OK, you IDL gurus. How do you delete an IDL system variable
> you have created with DefSysV without getting out of the IDL session?
>
> IDL> DefSysV, '!TESTER', 5
> IDL> DelVar, !TESTER
> % DELVAR: Expression must be named variable in this context: <INT
> (    5)>.
>
> IDL> Undefine, !Tester
> IDL> Help, !Tester
> <Expression>  INT      =      5
>
> My current thinking is that it is not possible. Any good ideas?
>
> Cheers,
>
> David
>
```

You are right, it is not possible:

"System variables have a predefined type and structure that cannot be changed." (IDL Reference Guide)

regards,
Lajos

Subject: Re: Tough System Variable Question for the IDL Gurus
Posted by [ben.bighair](#) on Thu, 12 Nov 2009 00:12:29 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Nov 11, 5:26 pm, FÖLDY Lajos <fo...@rmki.kfki.hu> wrote:

```
> On Wed, 11 Nov 2009, David Fanning wrote:
>> OK, you IDL gurus. How do you delete an IDL system variable
>> you have created with DefSysV without getting out of the IDL session?
>
>> IDL> DefSysV, '!TESTER', 5
>> IDL> DelVar, !TESTER
>> % DELVAR: Expression must be named variable in this context: <INT
>> (    5)>.
>
```

```
>> IDL> Undefine, !Tester
>> IDL> Help, !Tester
>> <Expression>  INT    =    5
>
>> My current thinking is that it is not possible. Any good ideas?
>
>> Cheers,
>
>> David
>
> You are right, it is not possible:
>
> "System variables have a predefined type and structure that cannot be
> changed." (IDL Reference Guide)
>
> regards,
> Lajos
```

Hi,

Why do you want to remove a system variable?

Would making the system variable a pointer or object work? At least then you could destroy/dereference them. I recall that I used an object as a system variable for the old "CoolHelp" thingy. And another thing, you *are* an IDL guru.

Cheers,
Ben

Subject: Re: Tough System Variable Question for the IDL Gurus
Posted by [David Fanning](#) on Thu, 12 Nov 2009 01:45:20 GMT
[View Forum Message](#) <> [Reply to Message](#)

ben.bighair writes:

> Why do you want to remove a system variable?

I don't know. I don't always come up with the questions.

> Would making the system variable a pointer or object work? At least
> then you could destroy/dereference them. I recall that I used an
> object as a system variable for the old "CoolHelp" thingy. And
> another thing, you *are* an IDL guru.

Yeah, but a guru who didn't know the answer. :-)

Maybe a pointer **will** work.

```
IDL> defsysv, '!tester', Ptr_New(5)
IDL> help, !tester
<Expression>  POINTER  = <PtrHeapVar795>
IDL> help, *!tester
<PtrHeapVar795> LONG    =          5
IDL> ptr_Free, !tester
IDL> !tester = Ptr_New(/ALLOCATE_HEAP)
IDL> help, /heap
Heap Variables:
  # Pointer: 1
  # Object : 0
```

```
<PtrHeapVar2>  UNDEFINED = <Undefined>
```

Thanks for the tip and the fish story, Ben! :-)

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Tough System Variable Question for the IDL Gurus

Posted by [David Fanning](#) on Thu, 12 Nov 2009 01:51:36 GMT

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

David Fanning writes:

```
> Maybe a pointer *will* work.
>
> IDL> defsysv, '!tester', Ptr_New(5)
> IDL> help, !tester
> <Expression>  POINTER  = <PtrHeapVar795>
> IDL> help, *!tester
> <PtrHeapVar795> LONG    =          5
> IDL> ptr_Free, !tester
> IDL> !tester = Ptr_New(/ALLOCATE_HEAP)
> IDL> help, /heap
> Heap Variables:
```

```
> # Pointer: 1
> # Object : 0
>
> <PtrHeapVar2> UNDEFINED = <Undefined>
```

Well, this could be confusing. Let me take my example all from the same IDL session:

```
IDL> defsysv, '!tester', Ptr_New(5)
IDL> help, !tester
<Expression>  POINTER  = <PtrHeapVar1>
IDL> help, *!tester
<PtrHeapVar1> LONG      =      5
IDL> Ptr_Free, !tester
IDL> !tester = Ptr_New(/Allocate_Heap)
IDL> help, !tester
<Expression>  POINTER  = <PtrHeapVar2>
IDL> help, *!tester
<PtrHeapVar2> UNDEFINED = <Undefined>
IDL> help, /heap
Heap Variables:
  # Pointer: 1
  # Object : 0
<PtrHeapVar2> UNDEFINED = <Undefined>
IDL> Ptr_Free, !tester
IDL> help, /heap
Heap Variables:
  # Pointer: 0
  # Object : 0
```

Cheers,

David

--

David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
Sepore ma de ni thui. ("Perhaps thou speakest truth.")
