
Subject: Structure in argument #2 -> With IDL ActiveX control

Posted by [Skiso](#) on Fri, 20 Nov 1998 08:00:00 GMT

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

Hi,

Does the GetNamedData() function of the IDL ActiveX control support the structure definition?

Example: MyVB_variable = IDLDrawWidget.GetNamedData("MyIDL_variable")

It doesn't seems to support the structure!

Skiso

Subject: Re: Structure in argument #2 -> With IDL ActiveX control

Posted by [Skiso](#) on Thu, 26 Nov 1998 08:00:00 GMT

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

I understand that it is impossible to pass a entire structure from IDL to VB. But, why can't I just pass the value of one field of a structure like this:

Example: MyVB_variable = IDLDrawWidget.GetNamedData("MyIDL_structure.my_variable")

It would be _very_ useful!

Skiso

Mark Leaming wrote:

> On Mon, 23 Nov 1998 08:13:43 -0700, "Matthew J. Sheats"

> <sheats@lanl.gov> wrote:

>

>>> Does the GetNamedData() function of the IDL ActiveX control support the
>>> structure definition?

>>

>> Well, the GetNamedData() function returns data as a variant which needs
>> casting. If you have a user defined structure, then there isn't a
>> default cast for it, but what you might be able to do, is write a VB
>> function that could rip the Variant apart into the structure you need.

>>

>> The only issue is, I don't know how IDL would handle a request to
>> return a structure in this way. If it returns bits, then you're ok, but
>> if it doesn't know what to do.. you probably will just get an error.

>>

>> Let me know if you figure it out, I've never tried that before.
>>
>> Matt Sheats
>> Los Alamos National Laboratory
>
> GetNamedData will return an empty variant if the IDL variable that you
> are getting is a structure. The data types that you can retrieve
> using this function are scalars of type byte, short, long, float,
> double and string.
>
> =====
> Mark Leaming
>
> Research Systems, Inc.
> =====

Subject: Re: Structure in argument #2 -> With IDL ActiveX control
Posted by [davidf](#) on Sat, 28 Nov 1998 08:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Matthew J. Sheats (sheats@lanl.gov) writes:

> Definately seems to be a round about way of doing things. Funny thing
> is, I've seen similar behavior in places in IDL (not involving
> ActiveX). Like.. (and maybe this is just me being a moron), I can't do
> this:
>
> READU, Lun, Struct.X
>
> It doesn't seem to assign anything to Struct.X. What I always end up
> doing is something like this:
>
> READU, Lun, X
> Struct.X = X
>
> Pretty annoying. It's one of those things I never thought about
> reporting because I assumed it was a "feature" of some sort. One I
> could do without. Oh and ya, the structure and X and everything is
> defined prior to the READU.

This is because structure de-references are passed into procedures and functions (such as READU) by **value** and not by **reference**. The **ONLY** thing passed into IDL by reference are variables (whole variables, not parts of variables).

Here is an article on my web page with more detailed

information:

http://www.dfanning.com/tips/read_subscripted_array.html

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting

Phone: 970-221-0438 E-Mail: davidf@dfanning.com

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

Toll-Free IDL Book Orders: 1-888-461-0155

[Note: This follow-up was e-mailed to the cited author.]

Subject: Re: Structure in argument #2 -> With IDL ActiveX control

Posted by [Matthew J. Sheats](#) on Sat, 28 Nov 1998 08:00:00 GMT

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

> I understand that it is impossible to pass a entire structure from IDL to VB. But,
> why can't I just pass the value of one field of a structure like this:
>
> Example: MyVB_variable = IDLDrawWidget.GetNamedData("MyIDL_structure.my_variable")
>
> It would be _very_ useful!

> Skiso

Oh agreed. That is annoying. I've always had to assign to some variable and then retrieve the data like:

```
IDLDrawWidget.ExecuteStr( "x = MyIDL_structure.my_variable" )  
MyVB_Variable = IDLDrawWidget.GetNamedData( "x" )
```

Definately seems to be a round about way of doing things. Funny thing is, I've seen similar behavior in places in IDL (not involving ActiveX). Like.. (and maybe this is just me being a moron), I can't do this:

```
READU, Lun, Struct.X
```

It doesn't seem to assign anything to Struct.X. What I always end up doing is something like this:

```
READU, Lun, X  
Struct.X = X
```

Pretty annoying. It's one of those things I never thought about reporting because I assumed it was a "feature" of some sort. One I could do without. Oh and ya, the structure and X and everything is defined prior to the READU.

I have 3 activex related bugs still awaiting resolution right now. They didn't make it into 5.2, so now I have to wait at least until... what? 5.2.1? OH well.

Welp, I've babbled long enough, ttyl8r.

Matthew Sheats
Los Alamos National Laboratory

Subject: Re: Structure in argument #2 -> With IDL ActiveX control
Posted by [Skiso](#) on Sun, 29 Nov 1998 08:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

"Matthew J. Sheats" wrote:

```
>> I understand that it is impossible to pass a entire structure from IDL to VB. But,
>> why can't I just pass the value of one field of a structure like this:
>>
>> Example: MyVB_variable = IDLDrawWidget.GetNamedData("MyIDL_structure.my_variable")
>>
>> It would be _very_ useful!
>
>> Skiso
>
> Oh agreed. That is annoying. I've always had to assign to some
> variable and then retrieve the data like:
>
> IDLDrawWidget.ExecuteStr( "x = MyIDL_structure.my_variable" )
> MyVB_Variable = IDLDrawWidget.GetNamedData( "x" )
>
> Definately seems to be a round about way of doing things. Funny thing
> is, I've seen similar behavior in places in IDL (not involving
> ActiveX). Like.. (and maybe this is just me being a moron), I can't do
> this:
>
> READU, Lun, Struct.X
>
> It doesn't seem to assign anything to Struct.X. What I always end up
> doing is something like this:
>
> READU, Lun, X
```

> Struct.X = X
>
> Pretty annoying. It's one of those things I never thought about
> reporting because I assumed it was a "feature" of some sort. One I
> could do without. Oh and ya, the structure and X and everything is
> defined prior to the READU.
>
> I have 3 activex related bugs still awaiting resolution right now.
>

I'm working with IDL 5.1 and the most annoying bug is that my program that use IDL activeX run just one in Visual Basic. If I need to run again the program I need to close Visual Basic end re-open it. I hope this is fixed in 5.2!

Skiso

> They didn't make it into 5.2, so now I have to wait at least until...
> what? 5.2.1? OH well.
>
> Welp, I've babbled long enough, ttyl8r.
>
> Matthew Sheats
> Los Alamos National Laboratory

Subject: Re: Structure in argument #2 -> With IDL ActiveX control
Posted by [Matthew J. Sheats](#) on Mon, 30 Nov 1998 08:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

> I'm working with IDL 5.1 and the most annoying bug is that my program that use IDL
> activeX run just one in Visual Basic. If I need to run again the program I need to close
> Visual Basic end re-open it. I hope this is fixed in 5.2!

Well, they actually *almost* fixed it in 5.1.1. But it turned out that it had not been fixed afterall. Turns out, Visual Basic wasn't letting go of it's IDL handles or something or other (not releaseing resources or something) and that was causing the crash. 5.1.1 let you run a couple times before crashing again.

5.2 seems to have fixed it, but I still get some odd behavior from now and again. And I can prove that visual basic isn't "shutting down" IDL each time I exit my sample program. I have a class that keeps track of the number of times a certain member function is called, and between runs, it doesn't reset. It doesn't crash out from underneath me anymore, but sometimes when I am trying to gracefully exit VB, it gives me some memory error and crashes.

Oh well, I guess baby steps are better than nothing.

Talk to you later,

Matthew Sheats
Los Alamos National Laboratory

Subject: Re: structure

Posted by [Pavel Romashkin](#) on Fri, 23 Apr 1999 07:00:00 GMT

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

```
> b=create_struct('A',findgen(10000),'B',findgen(10000))
>
> I like to have as result,
>
> b[10000].a
> b[10000].b
>
```

how about

```
temp = replicate({a: 0.0, b:0.0}, 10000)
```

```
temp.a = b.a
```

```
temp.b = b.b
```

```
b = temp
```

Hope this will do it.

Cheers,
Pavel
