
Subject: Invoking a COM object method, that results in two arrays

Posted by [Karlo Janos](#) on Fri, 11 Jul 2008 16:32:38 GMT

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

Hello!

Currently I am successfully using a function of the library XtremeFringe (<http://www.iot.es/XtremeFringe/index.html>) in IDL 7.0 by instantiating a COM object and then invoking its method with the needed parameters. The corresponding code looks like the following:

```
-----  
; generate COM object  
MyXtremeLibObject =  
  OBJ_NEW('IDLcomIDispatch$CLSID$IDENTIFIER_FROM_WINDOWS_REGIS TRY')  
  
; invoke the function  
resulting_array = MyXtremeLibObject -> THE_FUNCTION(some, para, meters)  
-----
```

This works for a function, that results in just one single array.
But in the case of a function resulting in two arrays it does not.

```
-----  
# invoke the method  
result2 = MyXtremeLibObject -> THE_SECOND_FUNCTION(some, para, meters)  
-----
```

In this case the variable result2 has only two elements with zeros in it:

```
result2[0] => 0.0  
result2[1] => 0.0
```

Maybe these elements could be something like pointers to the addresses of arrays, but I did not find a way to use it in IDL.

In Matlab one gets the results in a cell array by something like

```
-----  
result=XFLib.The_Function(some, para,meters);  
resultarray1=result{1};  
resultarray2=result{2};  
-----
```

In IDL I do not know how to do it in a similar way.

I would be happy, if someone can give me a hint for solving my problem.

Best regards

Karlo

Subject: Re: Invoking a COM object method, that results in two arrays

Posted by [rtowler](#) on Fri, 18 Jul 2008 18:29:00 GMT

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

Karlo Janos wrote:

> Hello!

>

> Currently I am successfully using a function of the library
> XtremeFringe (<http://www.iot.es/XtremeFringe/index.html>) in IDL 7.0 by
> instantiating a COM object and then invoking its method with the needed
> parameters.

<snip>

> This works for a function, that results in just one single array.
> But in the case of a function resulting in two arrays it does not.

<snip>

> In Matlab one gets the results in a cell array by something like

>

> -----

> result=XFLib.The_Function(some, para,meters);

> resultarray1=result{1};

> resultarray2=result{2};

> -----

>

> In IDL I do not know how to do it in a similar way.

What are the types of the data in each of the cell arrays?

If they are different then there is no way you'll be able to use IDL's COM interface. The problem is that IDL doesn't have an equivalent to COM's VARIANT type. MATLAB's cell type is equivalent to COM's VARIANT which works out quite well for MATLAB.

I would love to see ITT extend IDL's COM interface to support VARIANT types via pointer arrays. I have a nearly complete ADO/COM based database class written but as long as you can only return one data type at a time it is fairly useless. I have a feature request in but I'm not holding my breath.

-Rick

N 062 00.12
E 179 51.60

Subject: Re: Invoking a COM object method, that results in two arrays
Posted by [Karlo Janos](#) on Fri, 18 Jul 2008 20:54:21 GMT

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

rtowler@gmail.com wrote:

> What are the types of the data in each of the cell arrays?

As far as I know the function results in two "double"-arrays. At least both arrays should be of the same type.

> If they are different then there is no way ...

And if they are of the same type? What chance do I have?

If there is no direct possibility to get the result, my last chance might be using a wrapper function written in C, that puts the results in one single array and can be invoked from IDL. Unfortunately I am not experienced in C, so the direct way would be the better one for me.

Thank you for your hints!

Karlo

Subject: Re: Invoking a COM object method, that results in two arrays
Posted by [rtowler](#) on Sat, 19 Jul 2008 00:52:50 GMT

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

On Jul 18, 1:54 pm, Karlo Janos wrote:

> rtowler wrote:

>> What are the types of the data in each of the cell arrays?

>

> As far as I know the function results in two "double"-arrays. At least
> both arrays should be of the same type.

>

>> If they are different then there is no way ...

>

> And if they are of the same type? What chance do I have?

>

> If there is no direct possibility to get the result, my last chance
> might be using a wrapper function written in C, that puts the results in

> one single array and can be invoked from IDL. Unfortunately I am not
> experienced in C, so the direct way would be the better one for me.

I'm guessing here, but your method probably returns a 2 element
variant array where each element is a variant containing doubles (in
other words a variant of variants). IDL doesn't handle variants
well. It can handle a variant that contains a single type, but it
cannot handle variants containing mixed types or as in your case
variants containing variants.

I think that you'll need to look at writing a dlm that acts as a
wrapper. If you want to win an award for the most obtuse route you
could use MATLAB's COM server functionality to pass this data to IDL:

```
ml=obj_new('IDLcomIDDispatch$PROGID$matlab.application')  
<insert MATLAB commands to invoke your COM client>  
null=ml->execute('result=XFLib.The_Function(some, para,meters);')  
resultarray1=ml->execute('result{1}')
```

```
resultarray2=ml->execute('result{2}')
```

Of course the data returned to IDL is of type string which you would
have to parse. I'm not saying it's a good idea :) If you decide to
write the DLM, you'll want Ronn Kling's book "Calling C and C++ from
IDL". I've found it indispensable. <http://www.kilvarock.com/books/callingCfromIDL.htm>

-Rick

N 061 12.39
W 179 47.38

Subject: Re: Invoking a COM object method, that results in two arrays
Posted by [Karlo Janos](#) on Sat, 19 Jul 2008 07:57:28 GMT
[View Forum Message](#) <> [Reply to Message](#)

rtowler@gmail.com schrieb:

> I think that you'll need to look at writing a dlm that acts as a
> wrapper. If you want to win an award for the most obtuse route you
> could use MATLAB's COM server functionality to pass this data to IDL:
> [...]

That looks nice ;), but unfortunately I don't have MatLab available on
the computer, where I use IDL.
Maybe I will try writing a wrapper routine. Thanks again for your advice!

Karlo
