

distinguishing javascript functions, objects using atl

Source: <http://www.tech-archive.net/Archive/VC/microsoft.public.vc.atl/2007-01/msg00198.html>

- *From:* "qdin" <ericjudin@xxxxxxxxxx>
 - *Date:* 16 Jan 2007 14:48:38 -0800
-

Not sure if this is the right group for this... feel free to suggest another.

Is there a way to distinguish a javascript function object from other javascript types, in ATL?

Some background:

I have a DOM containing a node of type IDispatchEx, for example Root.SomeDispatchExObject.

This DOM is constructed in ATL, and automation compliant, so the ActiveScript engine can access it.

In javascript I can attach any kinds of objects to this node, like so:

```
Root.SomeDispatchExObject.myString = "hello";  
Root.SomeDispatchExObject.myNumber = 7;  
Root.SomeDispatchExObject.callBack = function(){ doSomething(); }
```

and I can even run the callBack like this, as expected

```
Root.SomeDispatchExObject.callBack(); or  
Root.SomeDispatchExObject.callBack.call();
```

Now, in ATL, after the javascript engine has run the script, I traverse the whole DOM via IDispatch's GetTypeInfo api, and can see these new objects.

What I want to do:

I want to be able to figure out from ATL that "callBack" is a function object.

The problem is that when i call GetTypeAttr() on its IDispatchEx object, it reveals 0 funcs. (the cFuncs and cVars fields of the TYPEATTR struct are both 0).

I've tried QIing for JSGlobal::FunctionInstance, but that didn't work either.

I've also tried GetIDsOfNames on the function "call" and then "toString" (both of which, I believe, should be supported by javascript objects) but that didn't work either.

distinguishing javascript functions, objects using at

Can anyone help?

thanks,

Eric

.