

ATL 8 connection point compilation problem

Source: <http://www.tech-archive.net/Archive/VC/microsoft.public.vc.atl/2006-08/msg00494.html>

- *From:* "Amit" <amitjain27@xxxxxxxxxx>
 - *Date:* 30 Aug 2006 09:14:28 -0700
-

Hi

I was working on upgrading VS.NET 2003 Attributed COM modules to VS.NET 2005. Some of the modules implement connection points (event_source). I have run into a compilation problem when I moved to the new compiler. I get the following error when I

error C2679: binary '=' : no operator found which takes a right-hand operand of type 'ITestEventSource *' (or there is no acceptable conversion)

```
c:\program files\microsoft visual studio
8\vc\platformsdk\include\oidl.h(489): could be 'tagVARIANT
&tagVARIANT::operator =(const tagVARIANT &)'
while trying to match the argument list '(VARIANT,
ITestEventSource *)'.
```

My interfaces are defined as the following:

```
// ITestEventSource
[
object,
uuid("BF6C7E0F-F3DB-46F8-B578-50FB6CE2BA0D"),
helpstring("ITestEventSource Interface"),
pointer_default(unique)
]
__interface ITestEventSource : IUnknown
{
[propget, id(1), helpstring("property Test")]
HRESULT test([out, retval] DOUBLE * pVal);
};

// _ITestEventSourceEvents
[
dual, object, uuid("FB0A5A59-32EA-4C92-A56B-2B58C1083191"),
helpstring("_ITestEventSourceEvents Interface")
]
__interface _ITestEventSourceEvents : IDispatch
{
```

ATL 8 connection point compilation problem

```
[id(1), helpstring("EventFired")]  
HRESULT EventFired([in]ITestEventSource *pEditor);  
};
```

These interfaces are from a test application that demonstrates the errors as my real module is way too big.

The thing is that if I change `_ITestEventSourceEvents` to be a custom interface (i.e., not dual and does not derive from `IDispatch`), then the module compiles. The module also compiles if I change `ITestEventSource` to be dual (and derives from `IDispatch`) and leave the events interface `_ITestEventSourceEvents` as it is.

The reasons I do not want to make these kinds of changes are:

1. I am concerned if there will be binary incompatibility if I change the interface definition (by changing its type from custom to dual)
2. There are many modules that have similar structure and it would be a big task to go about changing all the modules.

Is this a bug in Microsofts ATL 8 or is this one of the standards conforming changes?

I will appreciate any ideas/suggestions on how to fix this problem.

Thanks
Amit

.