

Re: SoapClient reply transport issue

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.webservices.enhancements/2004-07/>

From: Lucien (*lucien*_at_MicrosoftAccount.com)

Date: 07/14/04

Date: Wed, 14 Jul 2004 12:42:52 -0700

Is the scenario that you want to send using RequestResponse and switch protocol for the reply on the same channel? How would that work? For instance if you use HTTP it will fail since HTTP will expect an HTTP response. If I misunderstood let me know.

If not on the same channel then you can change protocol using ReplyTo and it will be regarded as a different endpoint so you have to register an endpoint listening for that response. Also you have to use .SendOneWay() and set ReplyTo in the SoapEnvelope to the reply endpoint you're listening on. In this scenario you can have different protocols.

Lucien

"Craig Neuwirt" <cneuwirt@emsinet.com> wrote in message

news:%23V9IJbaEHA.3204@TK2MSFTNGP09.phx.gbl...

> *I want to reemphasize and issue that was raised by Roman Kiss in another post since no confirmation of the problem was identified.*

>

> *The WSE Messaging allows for a SoapClient to send a request on one transport*

> *and expect the reply using a different transport. This is done using the*

> *Context.Addressing.ReplyTo information. Unfortunately, it appears that*

> *SoapClient (acutally SoapClientAsyncResult) erroneously retrieves the*

> *reply*

> *input channel using the endpoint of the SoapClient.*

>

> *[from SoapClientAsyncResult]*

> *ISoapTransport transport1 =*

> *SoapTransport.GetTransport(client.Destination);*

> *this._channel = transport1.GetInputChannel(reference1,*

> *SoapChannelCapabilities.None);*

>

> *This prevents the ability to ever receiver responses of a different*

> *transport and removes a powerful fesature of WS-Messaging. I believe it*

> *should do the following*

>

> *ISoapTransport transport1 =*

microsoft.public.dotnet.framework.webservices.enhancements: Re: SoapClient reply transport issue

> *SoapTransport.GetTransport(context.Addressing.ReplyTo);*
> *this._channel = transport1.GetInputChannel(reference1,*
> *SoapChannelCapabilities.None);*
>
> *Is this a confirmed problem or the expected behavior? Are there any*
> *workarounds for this?*
>
> *Thanks,*
> *Craig*
>
>
>