

Re: .Net Remoting TCP or HTTP?

Source: <http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework/2005-01/1268.html>

From: Ken Kolda (*ken.kolda_at_elliemae-nospamplease.com*)

Date: 01/26/05

Date: Tue, 25 Jan 2005 16:42:48 -0800

John's given a good description of why you would pick either the TCP or HTTP channel. So, I'll address the GenuineChannels (GC) portion.

One frequent use of remoting is to implement remote events, i.e. events which are fired on the server and received on the various clients. Unfortunately, in a WAN environment, remoting events don't function so well. The reason is that, when the server needs to notify the client of the event, it must make a TCP/HTTP connection to the client (i.e. the client acts as a server). In a WAN environment, this is often infeasible for a couple of reasons:

- 1) The client is behind a firewall which will not permit connections.
- 2) The client is on a separate network than the server and is using NAT. As a result, the server doesn't have a valid IP address with which to connect to the client.

GC resolves this by ensuring all connections are initiated from the client. In the TCP case, this means using a bi-directional channel, so that requests can go both from client to server and vice versa. In the HTTP case, the client actually creates two connections to the server — one used for for traffic in each direction. As a result, you can implement remote events in a WAN.

GC also adds a number of other features not available in native remoting without writing your own sinks, e.g.

- * Built-in encryption & compression options — another nice feature when used in a WAN.
- * Much greater configurability & reliability through tmeouts, connection retries, etc.
- * Event model on client and server to detect connections, reject connections, determine the client which is invoking the current method, etc.

If you download the GC demo from their web site, you'll get the documentation which describes all of these features plus many more that I've never used.

Hope that helps –
Ken

"John Puopolo" <john.puopolo@fastsearch.com.nospam> wrote in message
news:e1CTHCzAFHA.3772@TK2MSFTNGP10.phx.gbl...

> *Andre:*

>

> *In terms of using TCP directly or HTTP, the choice really revolves around*
> *the source system, the target system and what's in between. If the client*
> *and server are both .NET and you do not need to worry about firewalls in*
> *between them, use the TCP channel with a binary formatter (the default).*
> *This scheme will not work, however, if one of the endpoints is not .NET or*
> *if there are firewalls in between the endpoints. The firewalls (or some*
> *routers) will reject traffic not originating or destined to port 80.*

>

> *In these cases, you still have an "efficiency" option. You can send*
> *binarized data over the HTTP channel. This won't be quite as efficient as*
> *sending the information over the TCP channel directly, but it does offer a*
> *nice balance of speed and flexibility.*

>

> *I have not hsed the genuine channels product, but I have seen mention of*
it

> *around the remoting newsgroups. I believe it is a set of classes that*
> *simplifies using remoting channels, and implements "difficult to*
implement"

> *functionality such as remoting callbacks, etc. I am interested to see*
other

> *reponses to this part of you question as a way of learning a little bit*
> *more about what it does and perhaps how it works.*

>

> *Hope this helps,*

> *John Puopolo*

>

> *"André" <Andr@discussions.microsoft.com> wrote in message*
> *news:A80ED776-BE75-4AC9-B232-772E0F2F3D25@microsoft.com...*

>> *Hi,*

>>

>> *I'm new to .Net Remoting, I am wondering what should I use for my*
channel,

>> *TCP or HTTP.*

>>

>> *Can someone tell me why I would use one over the other? And in what*
cases

>> *will I use TCP over HTTP and vice versa*

>>

>> *I am using C# .net. I am behind a firewall; this app will be for a few*

>> *hundred users over a WAN, internal only. Better performance is what I'm*

>> *looking for.*

>>

>> *What I'm trying to build is a middle ware for my windows app's to*
connect

microsoft.public.dotnet.framework: Re: .Net Remoting TCP or HTTP?

> > *to, and the remoting server will talk to Oracle database and return the*
> *data*
> > *to the client.*
> >
> > *And. does anyone know about <http://www.genuinechannels.com/> what is*
this?
> > *And why would I need this?*
> >
>
>