

## Tip for Debugging Remoting Calls in Flex Builder 3

Posted At : August 21, 2009 4:27 PM | Posted By : Stefan Richter

Related Categories: Flex, ColdFusion, Tools

Imagine the times when **Flash Builder** (Gumbo) was still in beta, times when the **Network Monitor** was still only a twinkle in the corner of a Flex engineer's eye. Imagine... your current day job, and a Flex 3 project with a bit of ColdFusion powered Remoting (other Remoting gateways are available) thrown in. Which is going wrong. And you need to debug the Remoting calls.

Ok, that doesn't require much imagination, I know. We know that Flash Builder will make our lives somewhat easier when it finally ships, but in the meantime your boss doesn't allow you to run beta software so what does one do besides coming to work for me? And maybe, just maybe, your boss is really mean (unlike myself) and won't even pay out for a license of **Charles**. So now what?

I present to you . It's a little tag tat does wonders to your Flex Builder 3 Console, and I don't know why it isn't being promoted much more. I've been using it for quite a while, and while I do not recall how I discovered it I am sure it's never too late to tell the world all about it.

So what does **TraceTarget** do? In a nutshell it outputs a ton of logging information into your Console when you debug your Flex project, and in particular it tells you a lot about the Remoting calls that you are undoubtedly making. Here's an example of my current project, first without TraceTarget added:

```
[SWF] Users:stefan:Documents:dev:flex:fb3:demo:bin-debug:index.swf - 1,379,424 bytes after decompression
result getUserDetails
```

Yeah, nothing basically besides one trace that I out in there to signal the return from CF. Now let's add to the main.mxml file and debug the app again:

```
[SWF] Users:stefan:Documents:dev:flex:fb3:demo:bin-debug:index.swf - 1,383,632 bytes after decompression
'9670B130-68C5-3D75-D251-3D30C4A99F2E' producer set destination to 'ColdFusion'.
'F2AF48B3-896A-FA9D-D77A-3D30C4AF2A71' producer set destination to 'ColdFusion'.
'8495032E-7DE7-49B1-124E-3D30C4EB0643' producer set destination to 'ColdFusion'.
'8495032E-7DE7-49B1-124E-3D30C4EB0643' producer sending message '17E7B79A-6D4E-21C3-4F67-3D30C4EE538F'
'my-cfamf' channel endpoint set to http://www.demo.com/flex2gateway/
'my-cfamf' channel settings are:
<channel id="my-cfamf" type="mx.messaging.channels.AMFChannel">
  <endpoint uri="http://www.demo.com/flex2gateway/" />
  <properties>
    <polling-enabled>>false</polling-enabled>
  </properties>
</channel>
'my-cfamf' pinging endpoint.
'my-cfamf' channel is connected.
'my-cfamf' channel sending message:
(mx.messaging.messages::RemotingMessage) #0
  body = (Array) #1
    [0] "abcd"
  clientId = (null)
  destination = "ColdFusion"
  headers = (Object) #2
  messageId = "17E7B79A-6D4E-21C3-4F67-3D30C4EE538F"
  operation = "getRoomDetails"
  source = "api.v1.private.rooms"
  timestamp = 0
```

```
timeToLive = 0
'8495032E-7DE7-49B1-124E-3D30C4EB0643' producer connected.
'8495032E-7DE7-49B1-124E-3D30C4EB0643' producer acknowledge of '17E7B79A-6D4E-21C3-4F67-3D30C4EE538F'.
1
'9670B130-68C5-3D75-D251-3D30C4A99F2E' producer sending message '01B287E5-5EF5-1B40-75EE-3D30C648A026'
'9670B130-68C5-3D75-D251-3D30C4A99F2E' producer connected.
'my-cfamf' channel sending message:
(mx.messaging.messages::RemotingMessage)#0
  body = (Array)#1
  clientId = (null)
  destination = "ColdFusion"
  headers = (Object)#2
  messageId = "01B287E5-5EF5-1B40-75EE-3D30C648A026"
  operation = "getUserDetails"
  source = "api.v1.private.users"
  timestamp = 0
  timeToLive = 0
'9670B130-68C5-3D75-D251-3D30C4A99F2E' producer acknowledge of '01B287E5-5EF5-1B40-75EE-3D30C648A026'.
result getUserDetails
```

Pretty cool, right? You can read more about TraceTarget and other [logging frameworks here](#). But if you just want a quick and dirty way to see some more debug info on your Remoting calls then you could do worse than simply sticking the TraceTarget tag into your main.mxml file and save yourself some hassle. It may not be as slick as Charles or the upcoming Network Monitor but it can be handy nevertheless. Have fun coding.