

## Flex - Coldfusion Remoting, passing an Object usefully

Posted At : 1 July 2008 17:30 | Posted By : Shaun McCran  
Related Categories: Flex Remoting, Coldfusion, RIA, Flex

Ever noticed that with the increase in scale of a Flex application, the amount of data that you pass around seems to grow exponentially? Well I was fed up with passing multiple values back and forth from CF to flex, so after some digging here's the first article in using objects in remoting.

I want to be able to pass an object back and forth from CF to Flex, and if arguments are present use them, otherwise don't bother. This first article is based on constructing an object in flex, and receiving it in CF.

We send an object, but that's not what we get on the other end.....

### The CFC

First create your cfc, we are going to use cfproperty first to declare the types of our data.

```
<cfcomponent output="false">
    cfproperty name="county" displayname="county" hint="" type="string" default="" required="false"
    cfproperty name="tel" displayname="tel" hint="" type="string" default="" required="false"
    cfproperty name="country" displayname="country" hint="" type="string" default="" required="false"
```

Next we'll build our function, keeping the same arguments, in the same order. None of these are required, that way if they aren't in our object, then we just get an empty string.

```
<cffunction name="sendObject" access="remote">
    <cfargument name="county" type="string" required="false" default="">
    <cfargument name="tel" type="string" required="false" default="">
    <cfargument name="country" type="string" required="false" default="">
```

```
        county= #arguments.county#
        tel= #arguments.tel#
        country= #arguments.country#

    </cffunction>

</cfcomponent>
```

So that's our 'catching' cfc method. It uses named pairs that we will create in flex, as part of our object.

### The Flex

I'm going to assume that you know how to setup a remote object call to CF.

We create a function (sendMeStuff), which builds an object, and populates three arguments. Then we fire the object at the Remote Object method. This calls the ColdFusion cfc (catchMe) invoking the method 'sendObject'.

```
<mx:Script>
    <![CDATA[
        import mx.rpc.events.ResultEvent;
        import mx.controls.Alert

        private function sendMeStuff():void

            {
                var arguments:Object = new Object();
                arguments.county = "wiltshire";
                arguments.tel = "0800 49 59 69";
                arguments.country = "England";
                // fire it at your remote object
                RO.sendObject(arguments);
            }

        private function handleRO(e:ResultEvent):void

            {
```

```

        Alert.show('sent it')
        }

    ]]>
</mx:Script>

<mx:RemoteObject destination="ColdFusion"
                 id="RO"
                 fault="Alert.show(event.fault.message)"
                 source="catchme"
                 showBusyCursor="true">
:method name="sendObject" result="handleRO(event)" fault="Alert.show(event.fault.m

</mx:RemoteObject>
<mx:Canvas x="10" y="10" width="226" height="216" backgroundColor="#ffffff"
  <mx:Button x="67" y="79" label="Click me" click="sendMeStuff()"/>
  <mx:Text x="10" y="10" text="Click the button to send the stuff"/>
</mx:Canvas>

```

**Note:** We send in an object, but receive named pairs.

This seems like a much easier way of remoting than passing individual arguments about. I only have two reservations.

1. A colleague of mine has tried this through http service calls, IE as a webservice, and it doesn't seem to work. I will experiment and confirm....

2. CfProperty doesn't seem to make any difference at all! I've been investigating [Strongly typing data between flex and ColdFusion](#) and I was sure this was an example of this, but everyone else seems to think that cfProperty is crucial, anyone shed some light?

So now to work out the opposite! A cf object into Flex! (besides just setting datagrid.dataProvider = event.result)!