

Weather Web service (cfc) - Part 1

Posted At : 22 May 2008 13:28 | Posted By : Shaun McCran
Related Categories: Flex Remoting, Coldfusion, SQL, RIA, Flex

Recently I've been experimenting with consuming public Web Services and interpreting them through a Flex front end.

So I thought a handy remoting service would be a weather forecast feed consumer.

View the application [here](#) .

My first discovery was that due to sandbox security restrictions Flex isn't a great at interfacing with remotely hosted services.

This is mainly down to the crossdomain.xml policy.

Adobe has documented this 'feature' extensively [here](#) .

So to get around this I'm using a back end proxy service to do the WS calls, and then feeding the results into Flex, so to the CFC backend!

The Data

Before we look at the cfc code I'll mention that the data used in this is an industry formatted uk location code lookup table. Here's an example:

id	Code	Location Name
165	UKXX0011	Basingstoke
166	UKXX0012	Bath
167	UKXX0013	Batley

You can download the entire uk codelist and a sql creation script [here](#) .

The CFC

I've got into the habit of creating a new component per flex project, so create a new cfc, weather.cfc

In this cfc we have a standard cfc to flex combo population. We query a database, and do a query of queries to return lowercase data of the right name. (read more here).

```

function name="populateCombo" access="remote" hint="Populates the combo for uk locat:
    <cfquery datasource="#application.dsn#" name="propQuery">
        SELECT [varCode], [varLocation]
        FROM [dbo].[ukLocationCodes]
        order by varlocation
    </cfquery>

    <cfquery dbtype="query" name="qGetCombo">
        Select varCode as data, varLocation as label
        from propQuery
        Order by varLocation
    </cfquery>

    <cfreturn qGetCombo>
</cffunction>

```

This will populate a combo box in flex.

Next we have the actual http call to the yahoo weather service. Basically you take the url 'http://weather.yahooapis.com/forecastrss?p=' and append two different variables to it. The first is the uk location code. The second is optional, and specifies whether the return should be in Celsius or Fahrenheit. These are case sensitive, and should be a lower case 'c', or 'f'.

```

etForecast" access="remote" returntype="string" hint="Returns weather details for
    <cfargument name="locationCode" type="string" required="yes">
    <cfargument name="units" type="string" required="no">

    <cfset var url = 'http://weather.yahooapis.com/forecastrss?p='>
    <cfset url = url & arguments.locationCode>

    <cfif arguments.units NEQ "">

```

```
        <cfif arguments.units EQ 'c'>
            <cfset url = url & '&u=c'>
        <cfelseif arguments.units EQ 'f'>
            <cfset url = url & '&u=f'>
        </cfif>

        </cfif>

    <cfhttp url="#url#" method="get" resolveurl="yes">
        </cfhttp>

        <cfset returnVar = CFHTTP.FileContent>

        <cfreturn returnVar>
    </cffunction>
```

So that's Part 1 complete, now on to the flex that this populates!

Part 2 is [here](#) .