Passing an Array to a .NET Webservice

Posted At : April 27, 2009 11:09 PM | Posted By : Mark Kruger
Related Categories: ColdFusion

Here's a problem that will leave you scratching your head should you ever run into it. Consider a simple .NET web service that requires an "array of strings". The goal was to make use of a web service API published by Smarter Mail. I wanted a programmatic way of adding email aliases - groups of emails that function under a single address. The web service methods provided by the smarter mail API could not be simpler. Each request requires a username, password, domain and then additional stuff to make it work. For example, the "GetAlias()" function allowed me to pass in a domain and alias and get a list of emails already associated with that alias.

The problem came when it was time to add or update an alias. The argument for "addresses" to pass to the .NET service looked like "an array of strings" (that's how the help docs referred to is as well). The node in the XML looked pretty simple:

```
<AddAlias xmlns="http://tempuri.org/">
<AuthUserName>string</AuthUserName>
<AuthPassword>string</AuthPassword>
<DomainName>string</DomainName>
<AliasName>string</AliasName>
<Addresses>
<string>string</string>
<string>string</string>
</Addresses>
</AddAlias>
```

Now I can think of several ways to create an array of strings in ColdFusion so I started giving it the old college try. Unfortunately each attempt ended in failure. I could not figure out how to get a data type instantiated in CF to match the data type that .NET expected. I ended up experimenting with several different approaches to the array syntax.

After trying the straight up CF array syntax (as in x = arrayNew(1)), I tried JavaCasting to an array (seems obvious right?).

```
<Cfscript>
    Addresses = Javacast("string[]", ArrayOfAddresses);
</cfscript>
```

That didn't work so I tried directly instantiating a Java "arrayList" (as in createobject("java","java.util.ArrayList")), but that also yielded no results. I tried the "toArray()" member of the array class (based on this excellent post from Christian Cantrell on Passing ColdFusion Arrays to Java) but that was also a bust. I tried different ways of initing an array (like the bracket syntax). No matter what I tried I got the dreaded "argument type mismatch" error. Finally I stumbled onto this post on the Smarter Tools forum. A user with the same problem that I had created this function:

```
<cfscript>
    function SmartArray(lItems,delim){
var ArrayContainer=StructNew();
ArrayContainer.string=ListToArray(lItems,delim);
return ArrayContainer;
}
</cfscript>
```

What gives? His little function definitely solved my problem - but why? Looking carefully at the XML you can see why. The code is *not* calling for an array of string. It is calling for a member *named* "string" containing an array. If "addresses" was supposed to be an array of strings the node would look something like:

```
<Addresses>string</Addresses>
<Addresses>string</Addresses>
<Addresses>string</Addresses>
```

As it is written the array in question belongs not to "addresses" but to "addresses.string". My problem was getting hung up on the word "string" that was being used to label the node. I simply took it to be a type without recognizing it as an actual member of "addresses". All I needed was a wrapper "string" containing an array. Once I used this function I was able to work out the details of my Smarter Mail Web service and add a list of email addresses to an alias using code like this:

```
<cfscript>
    addresses = smartArray("email1@example.com,email2@example.com",",");
    mailServer.addAlias(AuthUsername=au, AuthPassword=pw, DomainName=domain,
AliasName=aliasName, addresses=addresses);
</cfscript>
```

To summarize, if your .NET service calls for an array of something, look carefully at the syntax and figure out the nature of the way the data is organized. Analyzing the XML carefully will give you a head start and keep you from puzzling till your puzzler is sore.