



Configuring a Data Quality SOAP Call Plan

This topic describes how to access a standard web service from a Data Quality (DQ) plan, which also includes a SOAP Call step. For more information on configuring DQ online services, see the How-to for *Creating, Testing, and Deploying a Data Quality Online Service*. For more information on configuring and using the SOAP Call step, refer to *DQ Tutorial 12.01 - SOAP Call*, which is included in iWay Data Quality Server (DQS).

This how-to includes the following components that are packaged in the *samples_soap.zip* archive:

- **.IN_HELLO.csv.metaData** – Sample metadata file.
- **JPSoapCallHello.plan** – Sample DQ plan file.
- **IN_HELLO.csv** – Sample input CSV file.

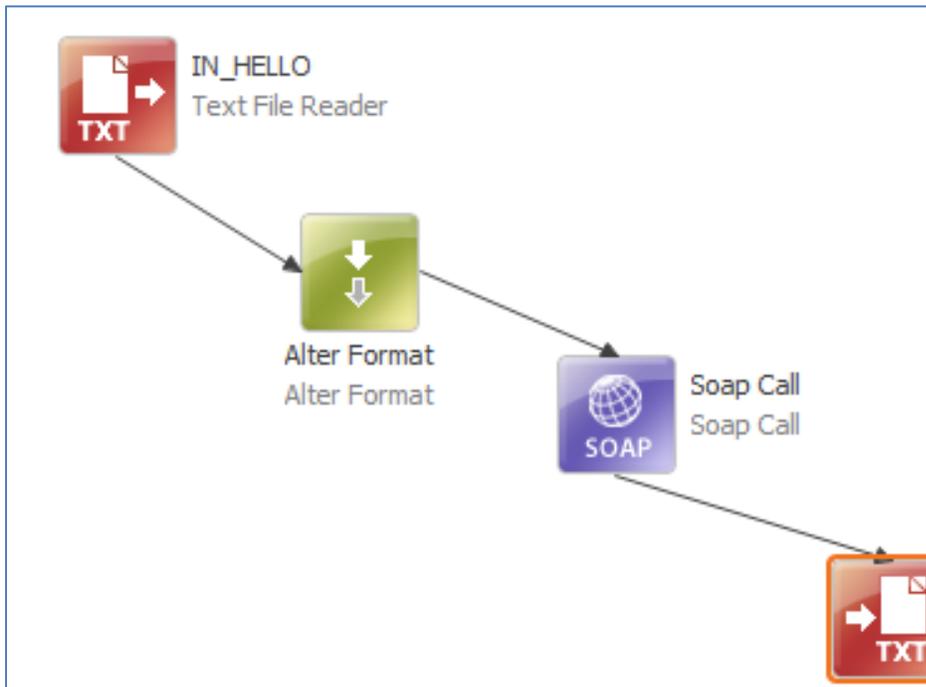
Before accessing a standard web service from a DQ plan, ensure your web service is online and available.

This how-to includes the following topics:

- [Creating a DQ Plan and Adding a SOAP Call Step](#)
- [Using a Text File Reader Step to Read the Input Name](#)
- [Using an Alter Format Step to Add Columns to Map the Response Output](#)
- [Configuring the SOAP Call Step](#)
- [Testing the SOAP Call](#)
- [Mapping the Output Columns](#)
- [Mapping an Input Field](#)
- [Running the SOAP Call Plan](#)

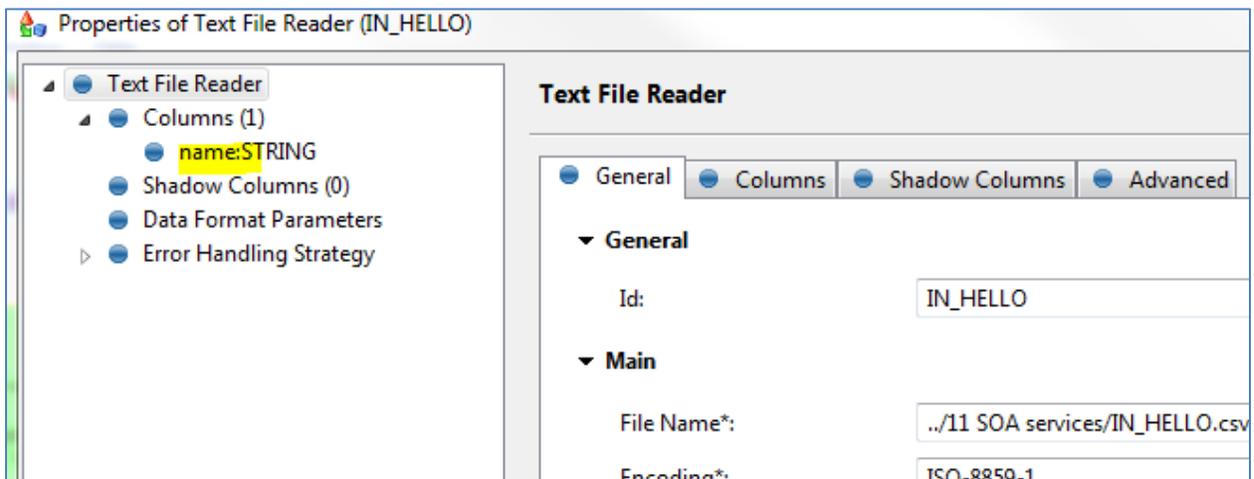
Creating a DQ Plan and Adding a SOAP Call Step

Using iWay Data Quality Server (DQS), create a new DQ plan and add a SOAP Call step, as shown in the following image.



Using a Text File Reader Step to Read the Input Name

Once you create a new DQ plan, add and configure a Text File Reader step for your DQ plan that will read the *name* column from your sample input data.

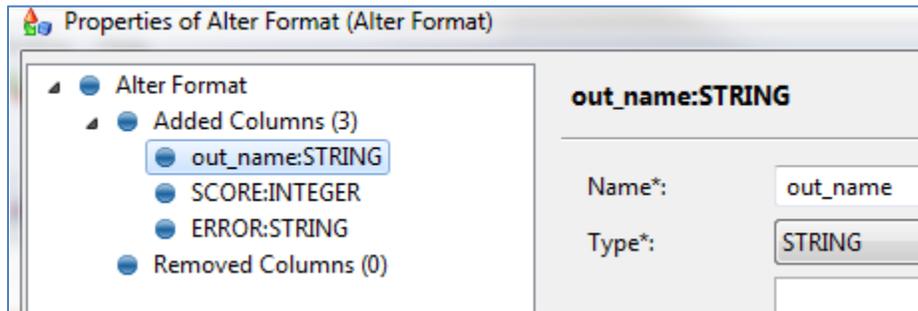


The following image shows the contents of a sample CSV document, which includes a name column with one row of data.

	name
1	Bond, James Bond

Using an Alter Format Step to Add Columns to Map the Response Output

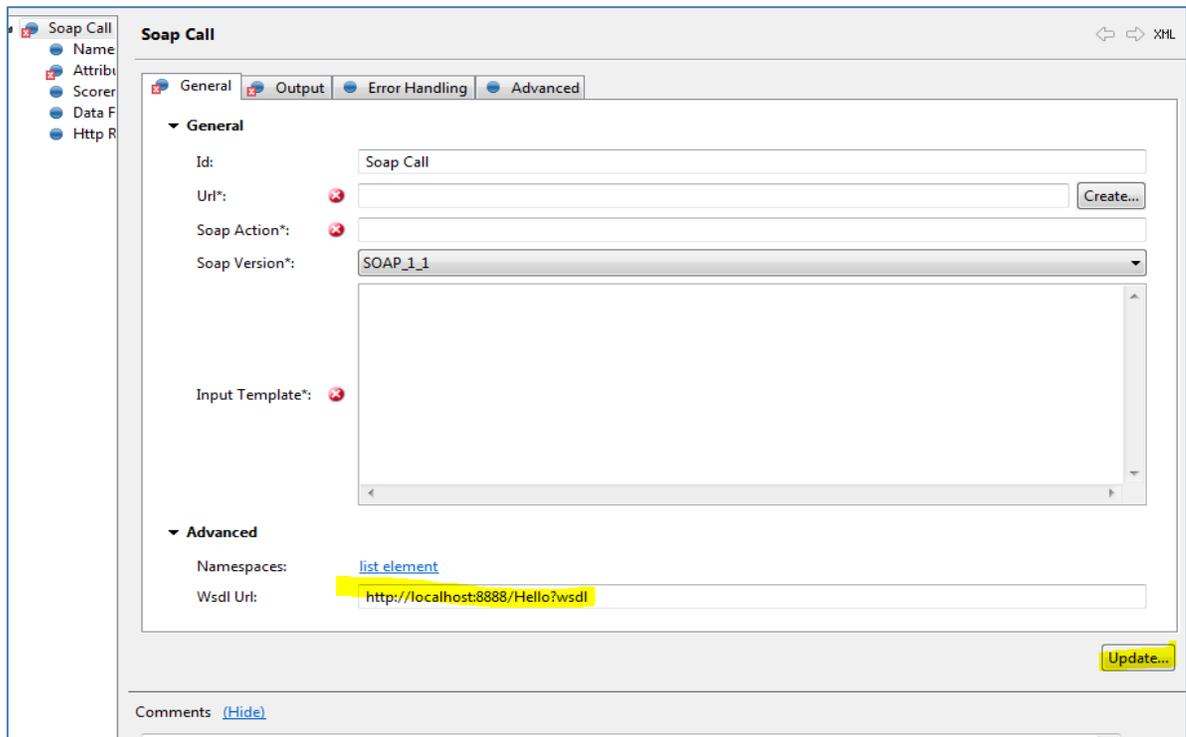
An Alter Format step can be used to add columns to map the response output, as shown in the following image.



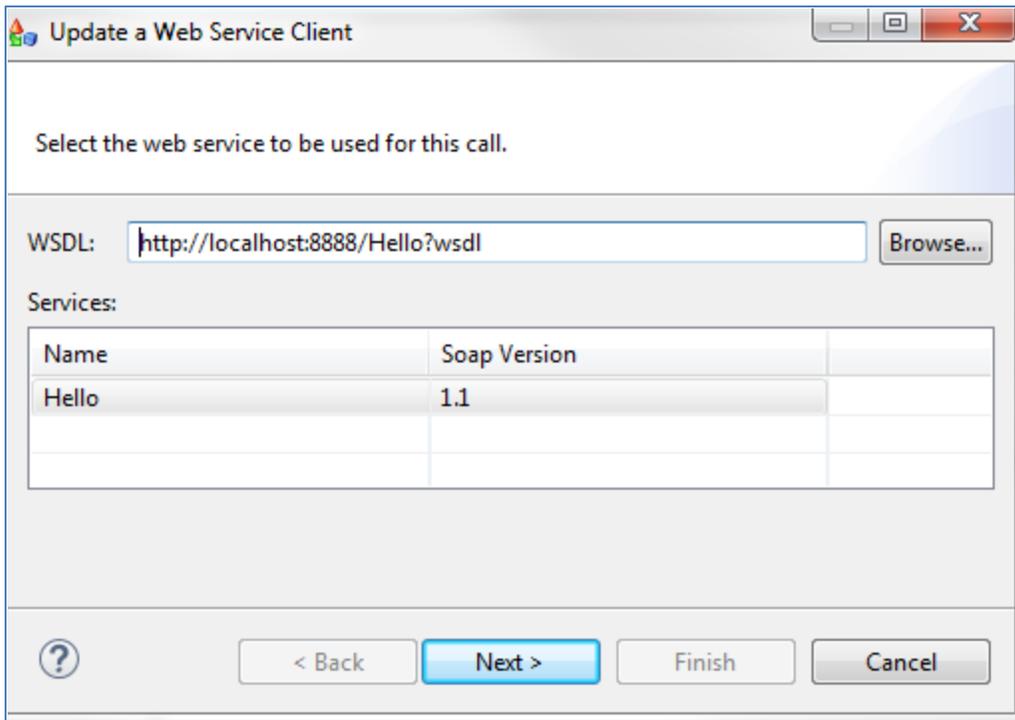
Configuring the SOAP Call Step

To configure the SOAP Call step:

1. Enter the WSDL URL (for example, <http://localhost:8888/Hello?wsdl>) in the corresponding field and click *Update*, as shown in the following image.



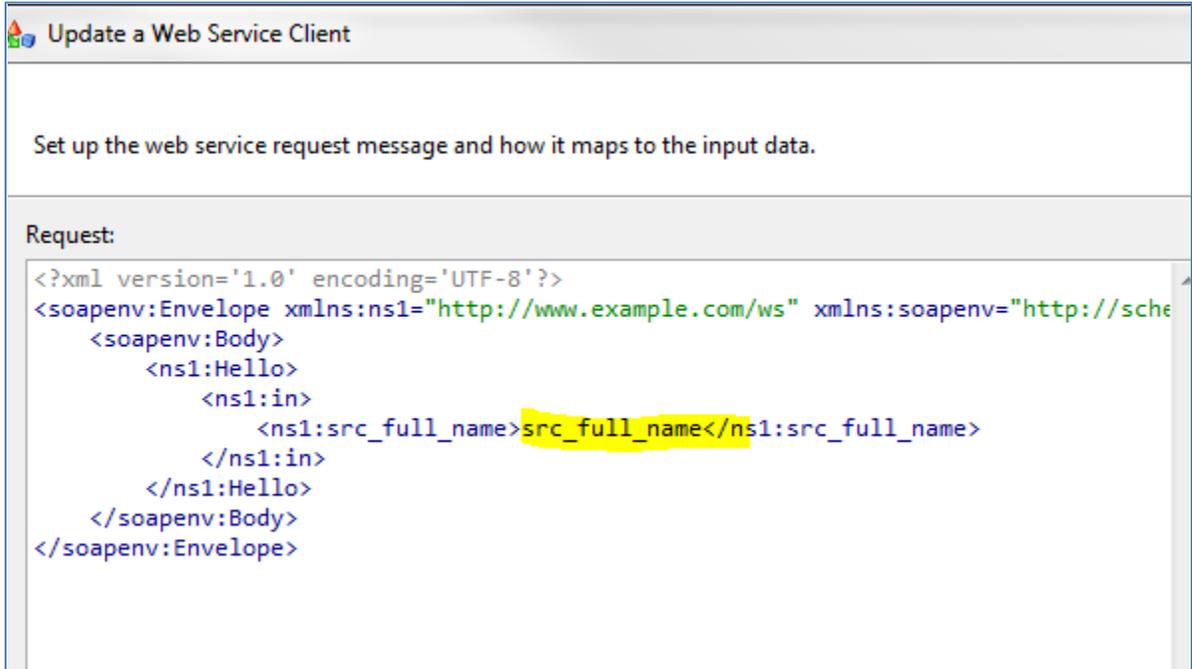
2. In the Update a Web Service Client dialog, select a service name from the Services list, as shown in the following image.



This will interrogate the live service and return its available functions.

3. Click *Next*.

The Request XML template appears with sample data, as shown in the following image.

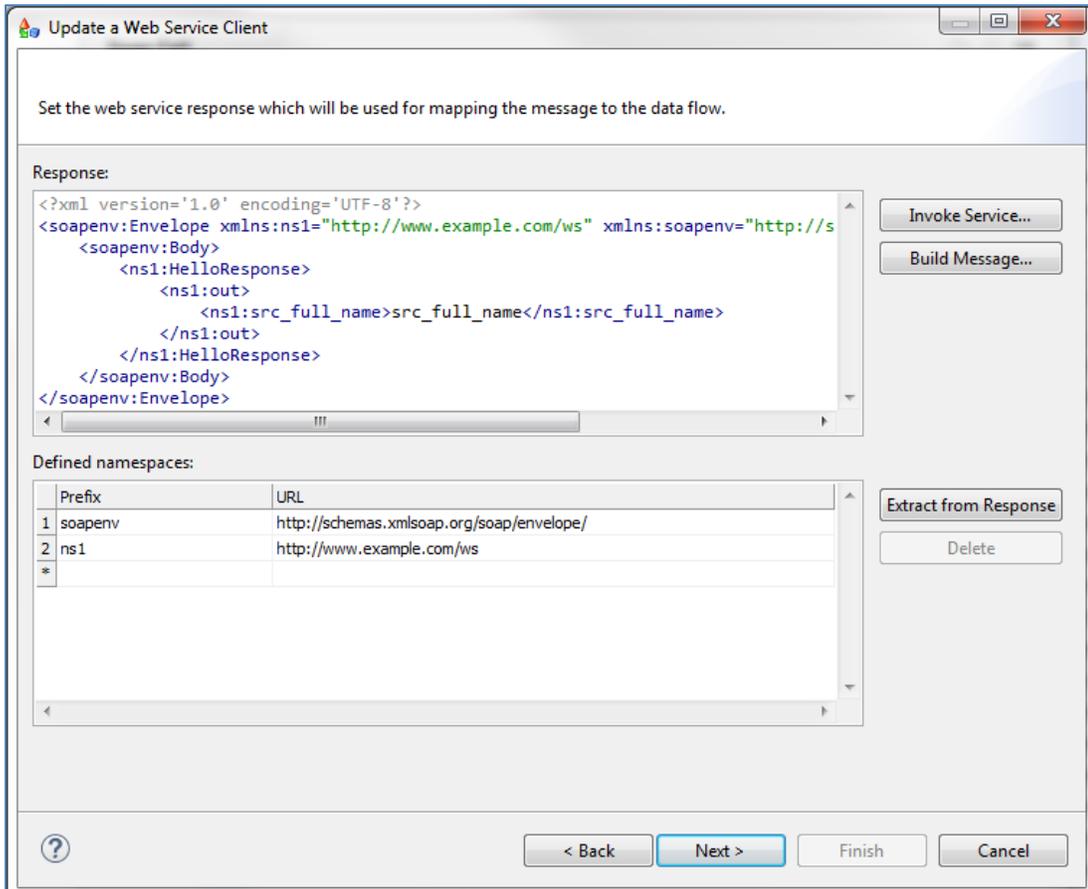


You can hard code a value or map a parameter from the input file.

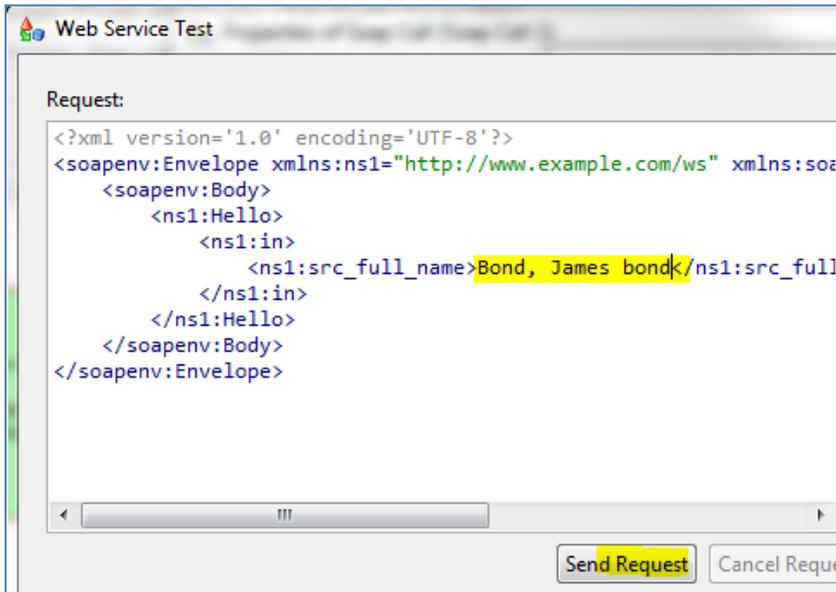
Testing the SOAP Call

To test the SOAP call:

1. In the Update a Web Service Client dialog, click *Invoke Service*, as shown in the following image.

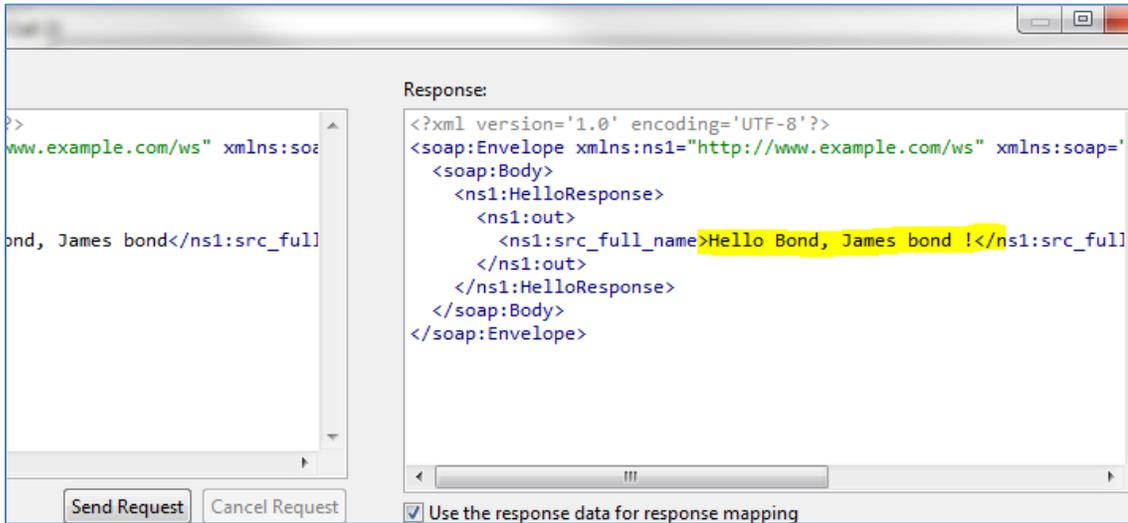


The Web Service Test dialog opens and displays the Request pane, as shown in the following image.



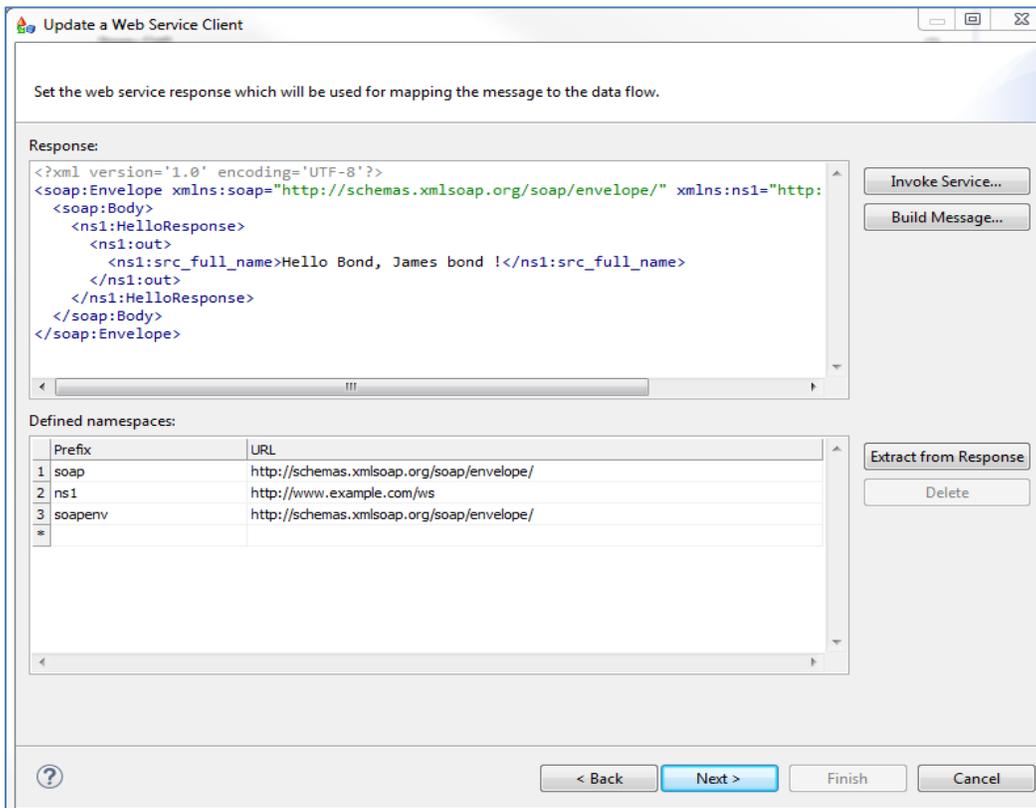
- Specify a value for the `<n1:src_full_name>` element (for example, type any name), and then click *Send Request*.

The response is returned, as shown in the following image.



- Click *OK*.

You are returned to the Update a Web Service Client dialog, as shown in the following image.

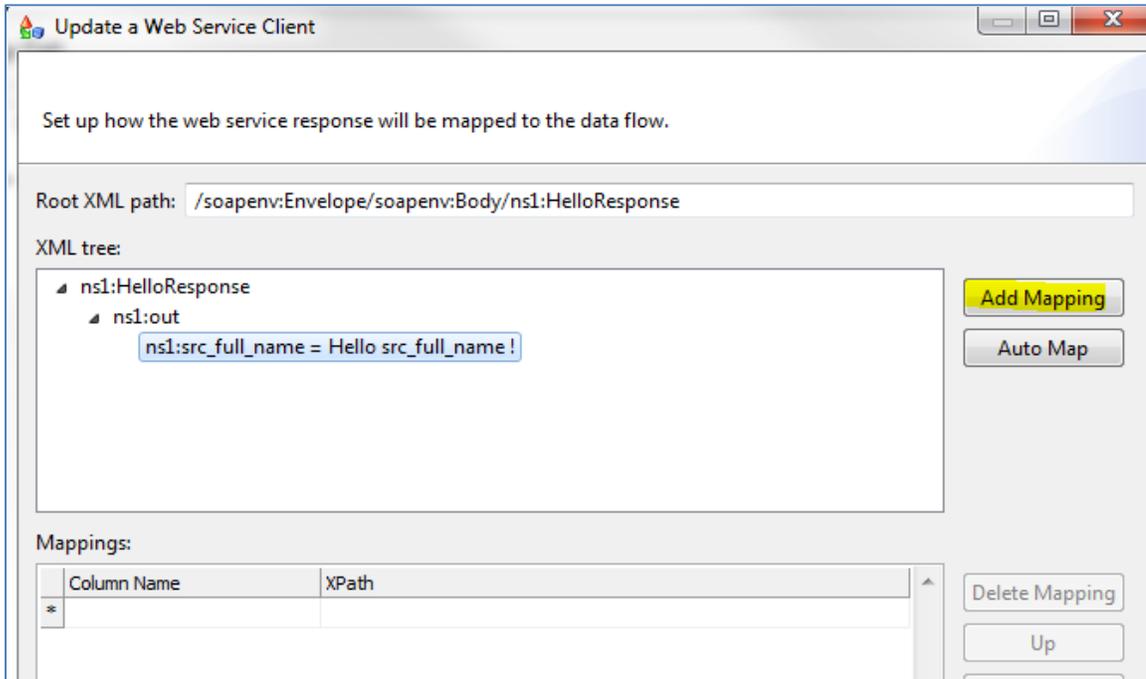


- Click *Next*.

Mapping the Output Columns

With the response now displayed, you must specify where it will go in your output. To map the output columns:

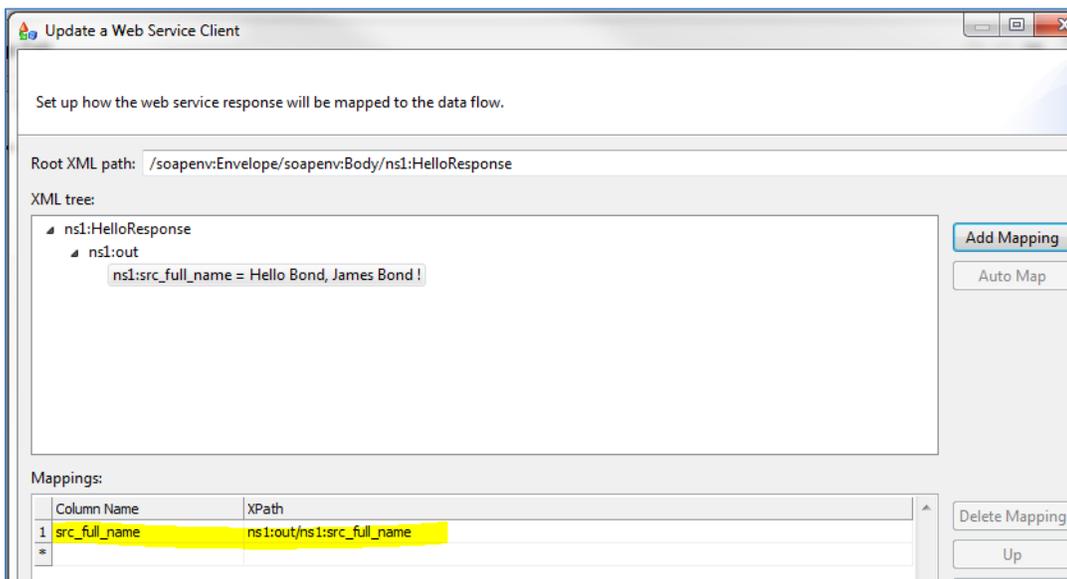
1. Select the response and navigate to the required field (for example, `src_full_name`), as shown in the following image.



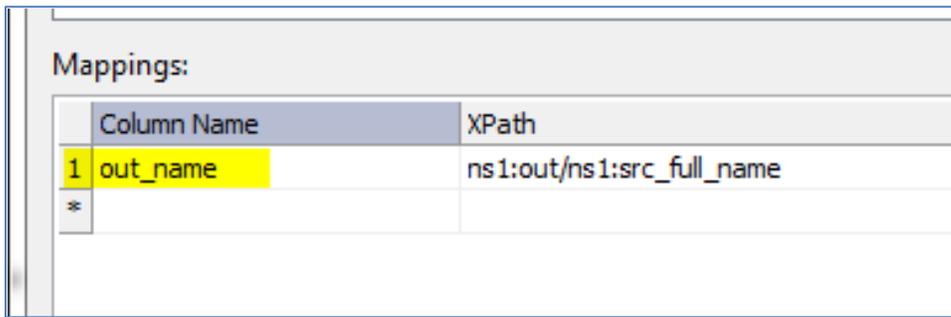
2. Click *Add Mapping*.

The XPath identifying the element that will be mapped into the output column is automatically inserted. For example:

```
/ns1:out/ns1:src_full_name
```

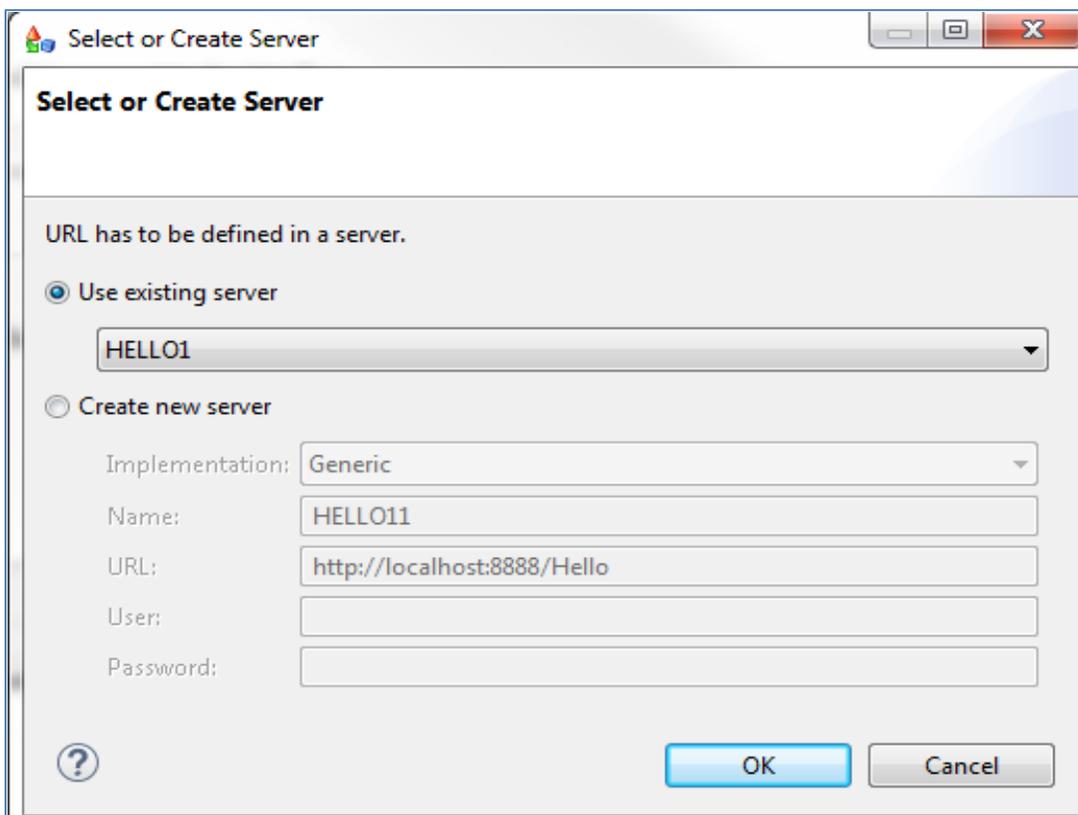


3. Change the output Column Name (for example, *out_name*), and then click *Finish*.



	Column Name	XPath
1	out_name	ns1:out/ns1:src_full_name
*		

The Select or Create Server dialog opens, as shown in the following image.



Select or Create Server

Select or Create Server

URL has to be defined in a server.

Use existing server

HELLO1

Create new server

Implementation: Generic

Name: HELLO11

URL: http://localhost:8888/Hello

User:

Password:

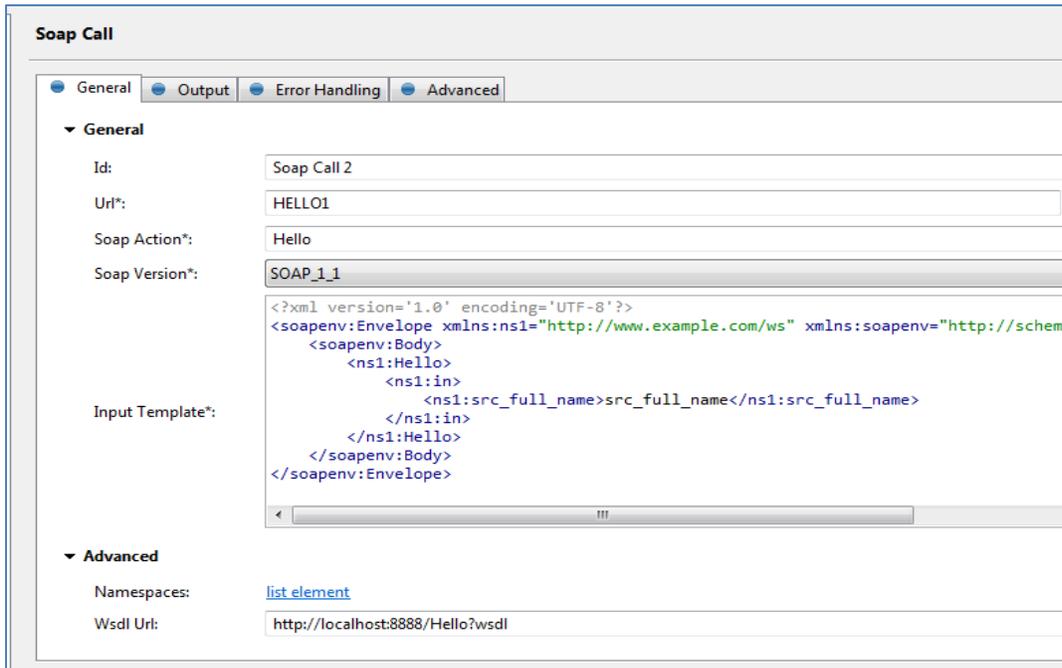
?

OK Cancel

4. Select *Use existing server* with the default value selected, and then click *OK*.

Mapping an Input Field

The configuration of the SOAP call plan is basically complete. However, you probably would like to have the option of changing the input every time you run the plan instead of reusing and resending the same hard-coded value.



The screenshot shows the 'Soap Call' configuration window with the 'General' tab selected. The configuration includes:

- Id:** Soap Call 2
- Url*:** HELLO1
- Soap Action*:** Hello
- Soap Version*:** SOAP_1_1
- Input Template*:** A text area containing the following XML code:

```
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope xmlns:ns1="http://www.example.com/ws" xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <ns1:Hello>
      <ns1:in>
        <ns1:src_full_name>src_full_name</ns1:src_full_name>
      </ns1:in>
    </ns1:Hello>
  </soapenv:Body>
</soapenv:Envelope>
```
- Advanced:** Namespaces: [list element](#); Wsdl Url: http://localhost:8888/Hello?wsdl

In the request template, you can use the `{in_column}` notation to specify a dynamic value. For example, the input file has a single column called *name*, as shown in the following image.



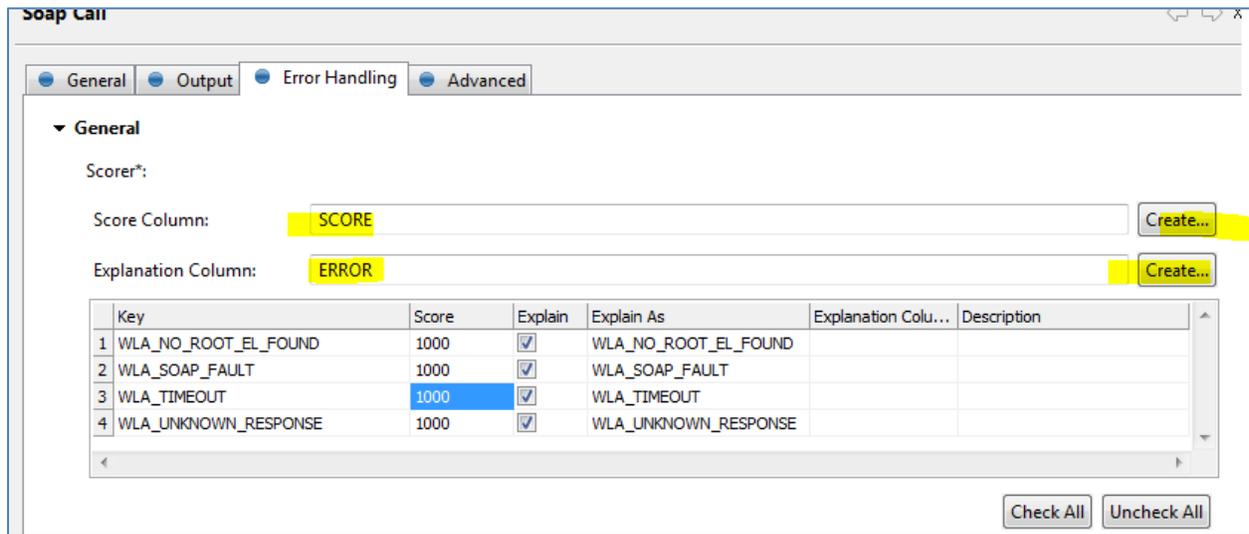
This close-up screenshot shows the 'Input Template*' field with the following XML code. The `{name}` placeholder is highlighted in yellow:

```
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope xmlns:ns1="http://www.example.com/ws" xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <ns1:Hello>
      <ns1:in>
        <ns1:src_full_name>{name}</ns1:src_full_name>
      </ns1:in>
    </ns1:Hello>
  </soapenv:Body>
</soapenv:Envelope>
```

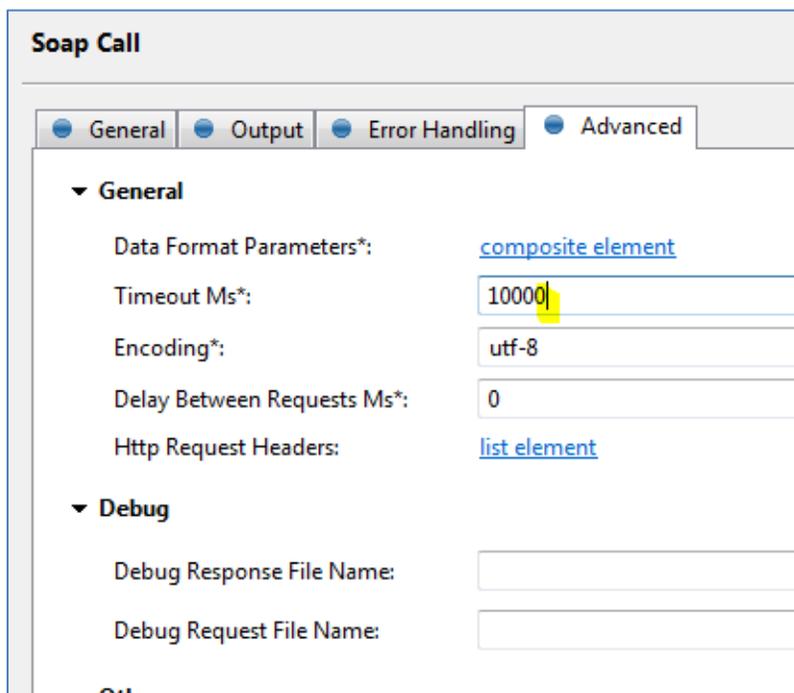
The value of this field will be mapped into the request input.

Running the SOAP Call Plan

Before you run the SOAP call plan, it is recommended that you create a new Score column and Explanation column, as shown in the following image.



1. In the Error Handling tab of the SOAP Call step, create a new Score column called *SCORE* and a new Explanation Column called *ERROR*. Click the *Create* button to add these columns if they have not already been added in the Alter Format step.
2. In the Advanced tab, set arbitrary scores so you can add logic to test for errors (for example, if a timeout occurs), as shown in the following image.



3. Increase the default timeout from 1 second (1,000 ms) to 10 seconds (10,000 ms), and then click *OK*.
4. Run the SOAP call plan.

The following output should be returned to the test plan. The input is concatenated with a greeting, as shown in the following image.

	name	out_name	SCORE	ERROR
1	Bond, James Bond	Hello Bond, James Bond !		

Any errors would be reflected in the SCORE and ERROR columns.