

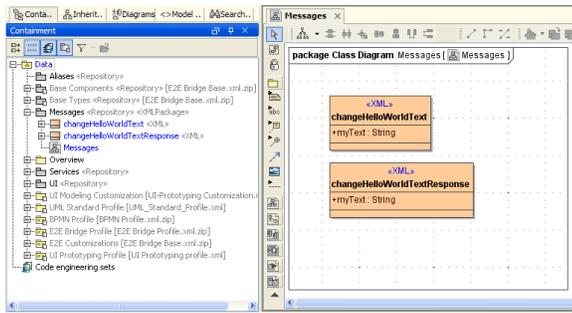
Defining the Service

The main functionality of this tutorial application is to replace a default text by a user input. Within xUML UI, the SOAP service with its messages represents the data model. The data model itself consists of the <<SOAPPortType>> operations input and output parameters. For best practice reasons, we use document/literal "wrapped" style messages (c.f. WS-I profile for reasons). This means, that each SOAP operation has at maximum one input and output parameter. The names of the parameter types are derived from the operation name: the input parameter has the same name as the operation, the output parameter is named by using the operation name having 'Response' added at the end.

Due to the fact that the SOAP operation parameter classes are not only used for the SOAP operation but at the same time are also used for the mapping of the data to the user interface elements, these classes are wrapped up in a **Messages** repository under the **Data** root element of the Containment tree.

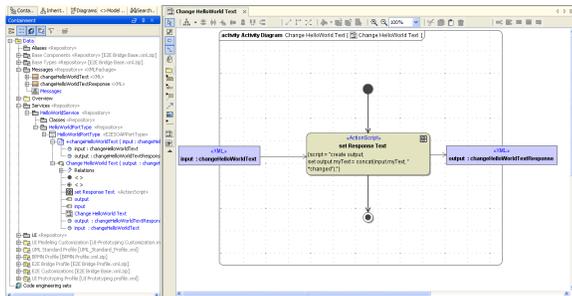


Defining the UI State Machine



As seen in the above figure, there are two classes, each with an attribute called **myText** of the BaseType String. The class **changeHelloWorldText** is the operations input parameter, the class **changeHelloWorldTextResponse** the output parameter.

Amend the default SOAP Service and change it to a SOAP Document operation implementing the above **changeHelloWorldText** and **changeHelloWorldTextResponse** classes as input and output parameters. Due to the nature of the implementation used for xUML UIs, SOAP operations must get the stereotype <<SOAPDocumentOperation>>. Finally, name the operation **changeHelloWorldText**.



For detailed information and a tutorial on how to create xUML SOAP Services refer to the [BRIDGE Development Tutorials](#).

To be able to view the user interface only, without having a valid data handling, it is possible to deploy the service without a SOAP service operation (meaning just the SOAP Port Type). This is an option if only the HTML user interfaces need to be presented or a specific design implementation (CI /CD) needs to be tested and functionality is not playing a role at that time.