T320 E-business technologies: foundations and practice

Block 3 Part 2 Activity 3: Editing the 'Hello' web service WSDL

Prepared for the course team by Neil Simpkins

Introduction

In this booklet I shall illustrate how Eclipse, with the WTP, supports WSDL creation and editing. I am going to use the WSDL document generated by Axis when the 'Hello' web service was created, but you can use any WSDL you choose.

If you want to use the 'Hello' service's WSDL and don't have the 'Hello' project to hand, you may want to redo the *Implementing a simple web service* activity that you carried out in Part 1. This can be completed in a very short time as you should now be familiar with the steps required.

The WSDL file generated by Axis in the 'Hello' project is called 'Hello.wsdl' and resides in the project's 'WebContent' folder, under 'wsdl' (Figure 1).
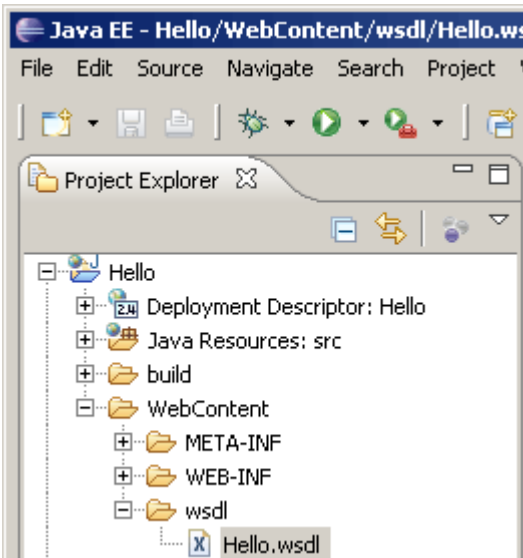
Figure 1 'Hello.wsdl' in the 'Hello' project

There are also some demonstration WSDL documents included with the T320 Eclipse distribution. To access these you can select File > New > Other..., then expand the 'Examples' folder and select 'Editing and validating XML files' (Figure 2). Click the 'Next' button and you will be taken to a dialogue box where you can give the project a name, or use the default 'XMLExamples' name, before clicking on 'Finish'.
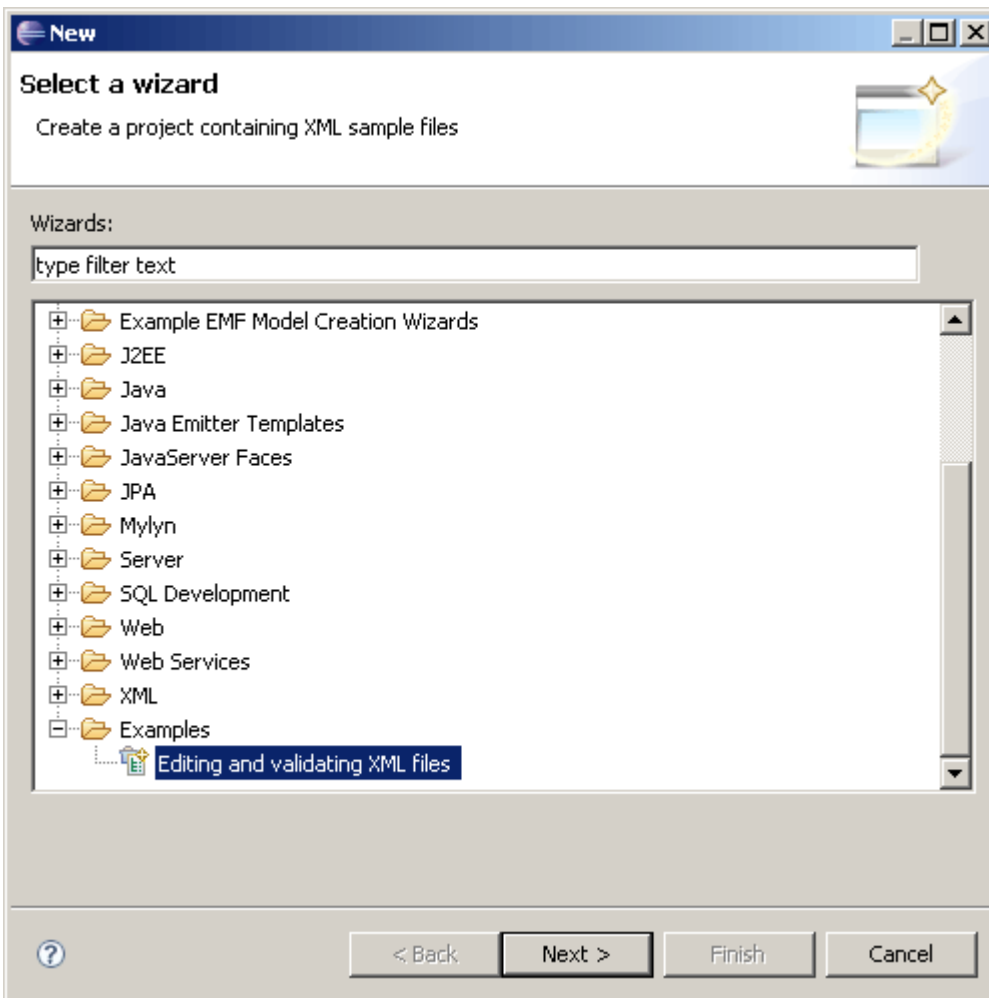

Figure 2 Creating a new project with examples

The examples contain two WSDL documents, one in the 'PhoneBanking' folder and the other in the 'SpaceWarGame' folder (Figure 3). If you do use the 'Hello' web service WSDL for this practical activity, as I shall here, it is also worth opening these WSDL files and examining them in the editor, as they illustrate slightly more complicated structures.
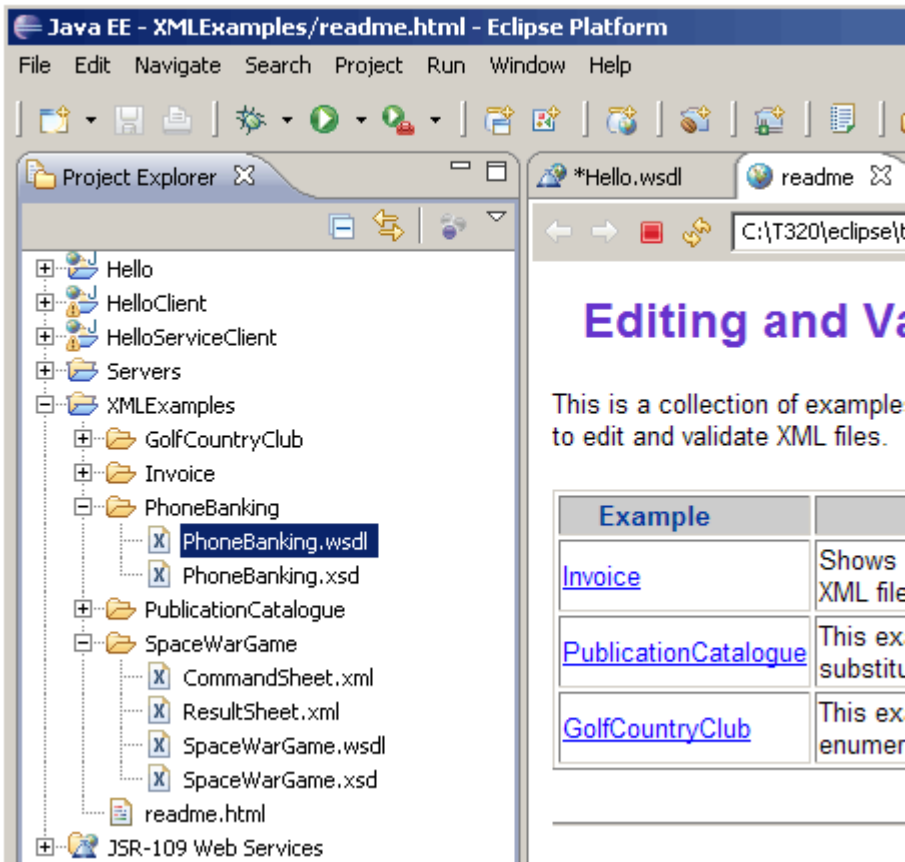

Figure 3   'XMLExamples' project expanded in the Project Explorer

Editing a WSDL document

If you double-click on a WSDL document, the file will be opened in the WTP WSDL editor in the centre of your workbench. This has two view options, available via tabs at the bottom left-hand corner of the panel: Design and Source. This is the same approach as for XML files; there is a graphical Design view, and a Source view that lists the XML of the WSDL document as plain text.

The WSDL document will automatically open with the Source view showing, but switch to the Design view. Your workbench should look something like Figure 4.
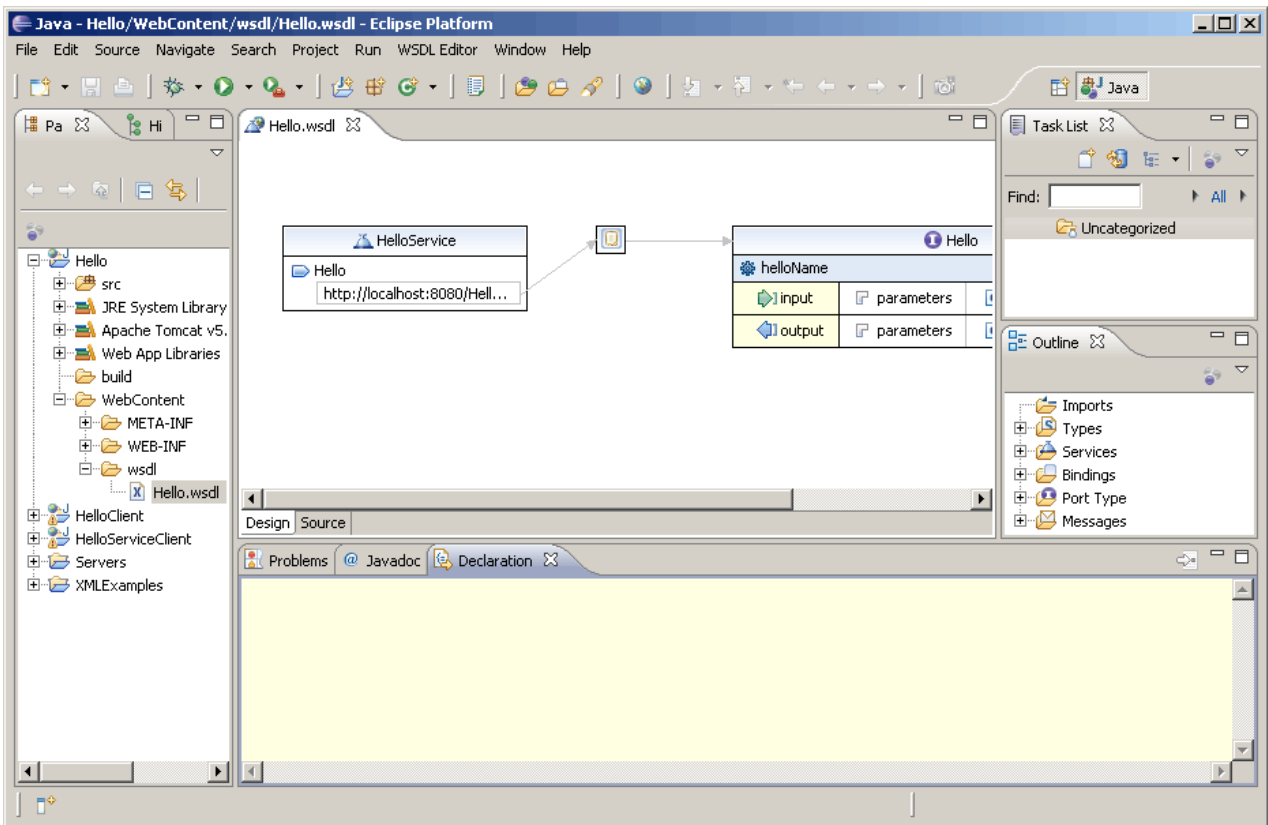
Figure 4 'Hello.wsdl' opened in the WSDL editor

The Design view consists of three boxes linked by arrows (Figure 5). From left to right, the three boxes represent:

1 the 'HelloService' service and its EPR, which is encoded in the WSDL within a port

2 a binding 'HelloSoapBinding', which connects the service to the portType and method descriptions

3 a list of the methods, such as 'helloName', that the service provides and for each of these a list of their input and output messages.

The first two items belong to the 'concrete' specifications, while the third (as you saw earlier) is more abstract.
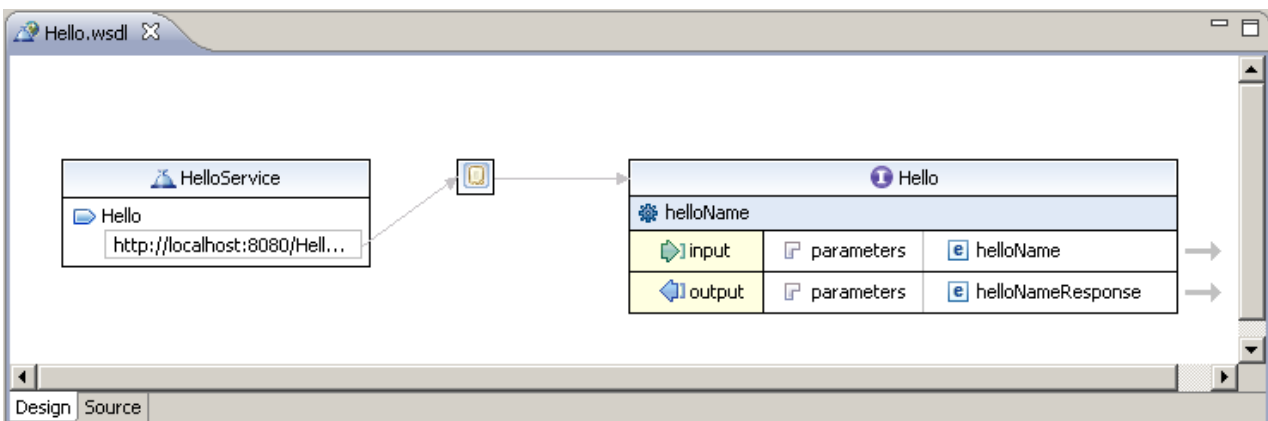

Figure 5 'Hello.wsdl' opened in the Design view

To the right of the Editor window, the Outline view lists six elements (Figure 6). You will see that five of these are the main elements of a WSDL document. The sixth element, 'Imports', allows, for example, namespaces to be imported.
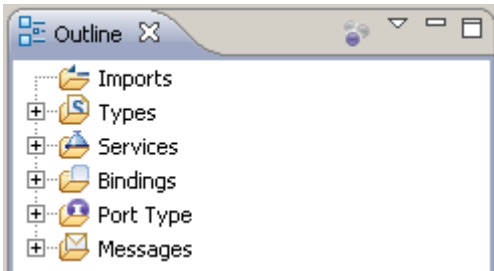

Figure 6 WSDL Outline panel in Eclipse

Returning to the Design view, the boxes and areas inside the boxes that make up the graphical view provide context-sensitive pop-up menus if you right-click. For example, right-clicking on the 'HelloService' label allows a new port to be added (Figure 7), while right-clicking on the binding icon allows the binding content to be edited or the portType to be changed (Figure 8).
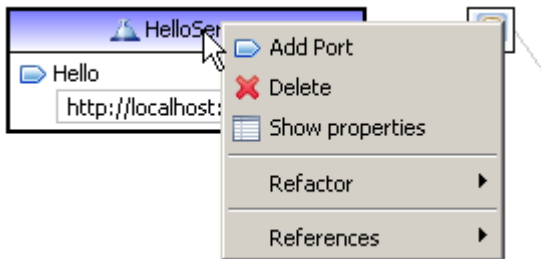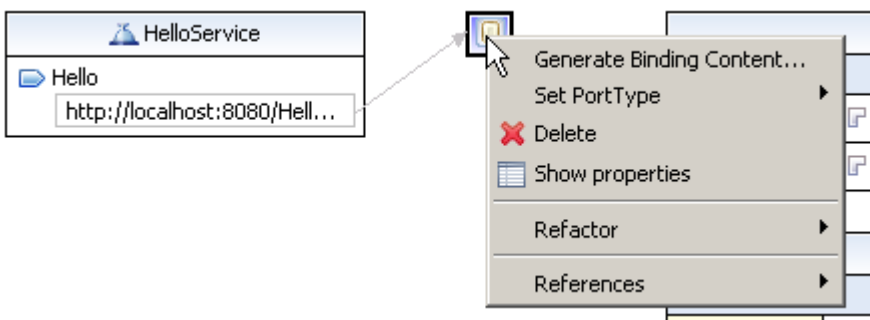

Figure 7 Service pop-up menu

Figure 8 Binding pop-up menu

The editor thus provides very good context- and syntax-sensitive editing facilities. It is also rather easy to create a WSDL from scratch if you need to do that.

One edit that you may need to make to a WSDL is to alter the endpoint. If you right-click on the 'HelloService' box where the 'Hello' port is shown as a URL and select Show properties (Figure 9) then, depending on your current perspective, a new Properties view will appear at the bottom of your Eclipse workbench or come to the front (Figure 10).
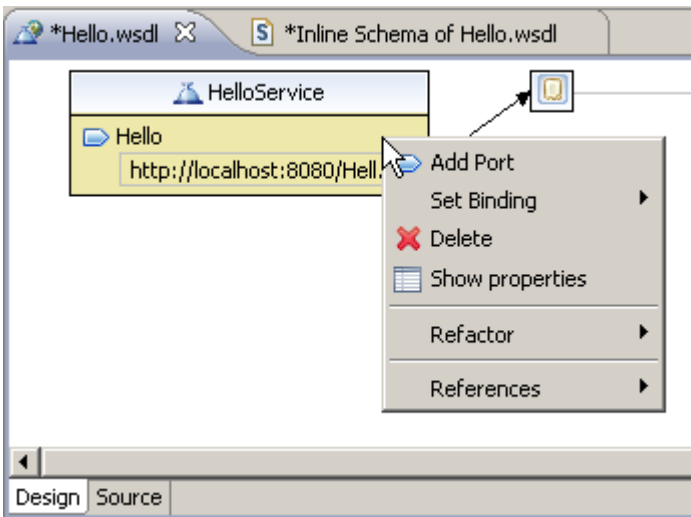
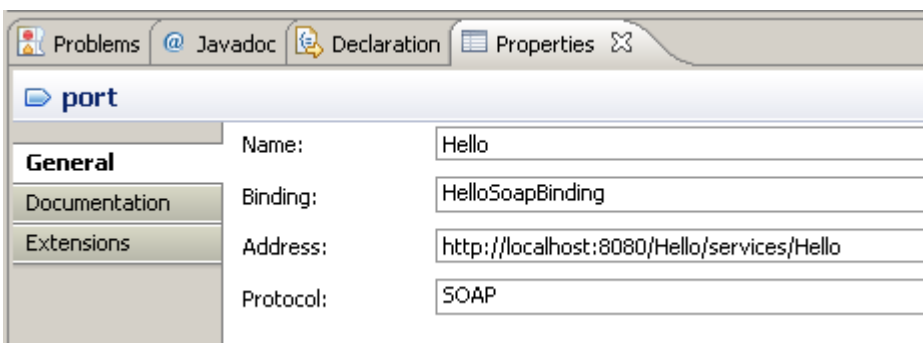**Figure 9** 'HelloService' port pop-up menu

Figure 10 'Hello' port properties

The port's properties include the address of the service, which can be changed to any value that is required if the service is moved onto another server.

You can examine and edit the properties of other components of the WSDL in a similar fashion. Take a look at the properties of other 'boxes' in the Design view. Once you have a Properties view showing in Eclipse, you can switch to the properties of other parts of the Design view simply by selecting the item in the diagram.

You may have noticed earlier that, in the Design view, to the right of the 'helloName' method box are two arrows pointing out from the box. If you hover your mouse over one of the arrows, it will turn blue and shortly a pop-up display will appear showing the details of the schema associated with the message (Figure 11). If you click on 'Open In New Editor', a new tab will appear in the main editing panel that is labelled 'Inline Schema of Hello.wsdl' (Figure 12).
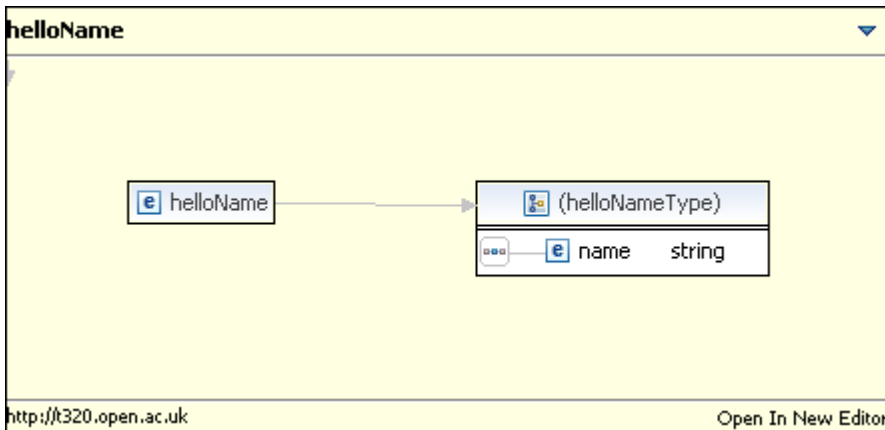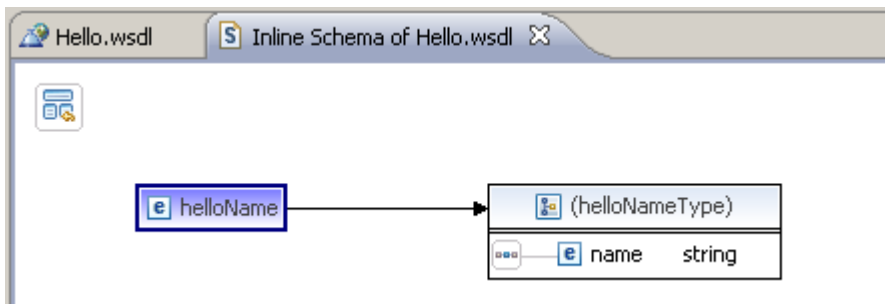
Figure 11 Pop-up schema display

Figure 12 Inline Schema editor

In Figure 12 the input to helloName is shown as having a single element called 'name', which is a string. Again the components of the diagram all support pop-up menus, which can be used to edit the diagram or show properties.

To see how you can edit the diagram, try to use pop-ups and/or the Properties view to change the WSDL to look like the one shown in Figure 13 (hint: I have added an element and then changed its type to 'boolean').
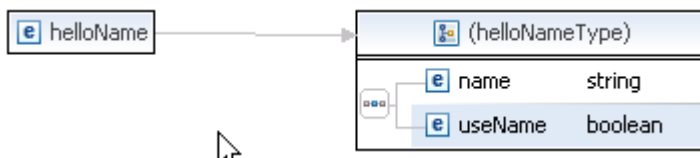

Figure 13 Edited 'helloName' input description

You might also have noticed that there is a small square button in the upper left-hand corner of the schema editor (Figure 14).
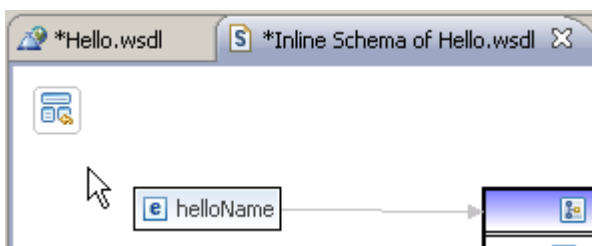

Figure 14 Schema Index view button

If you click this button you will switch to what is called the Schema Index view (Figure 15). This view lists all the items now shown in the Outline view, but in a slightly more graphical manner.
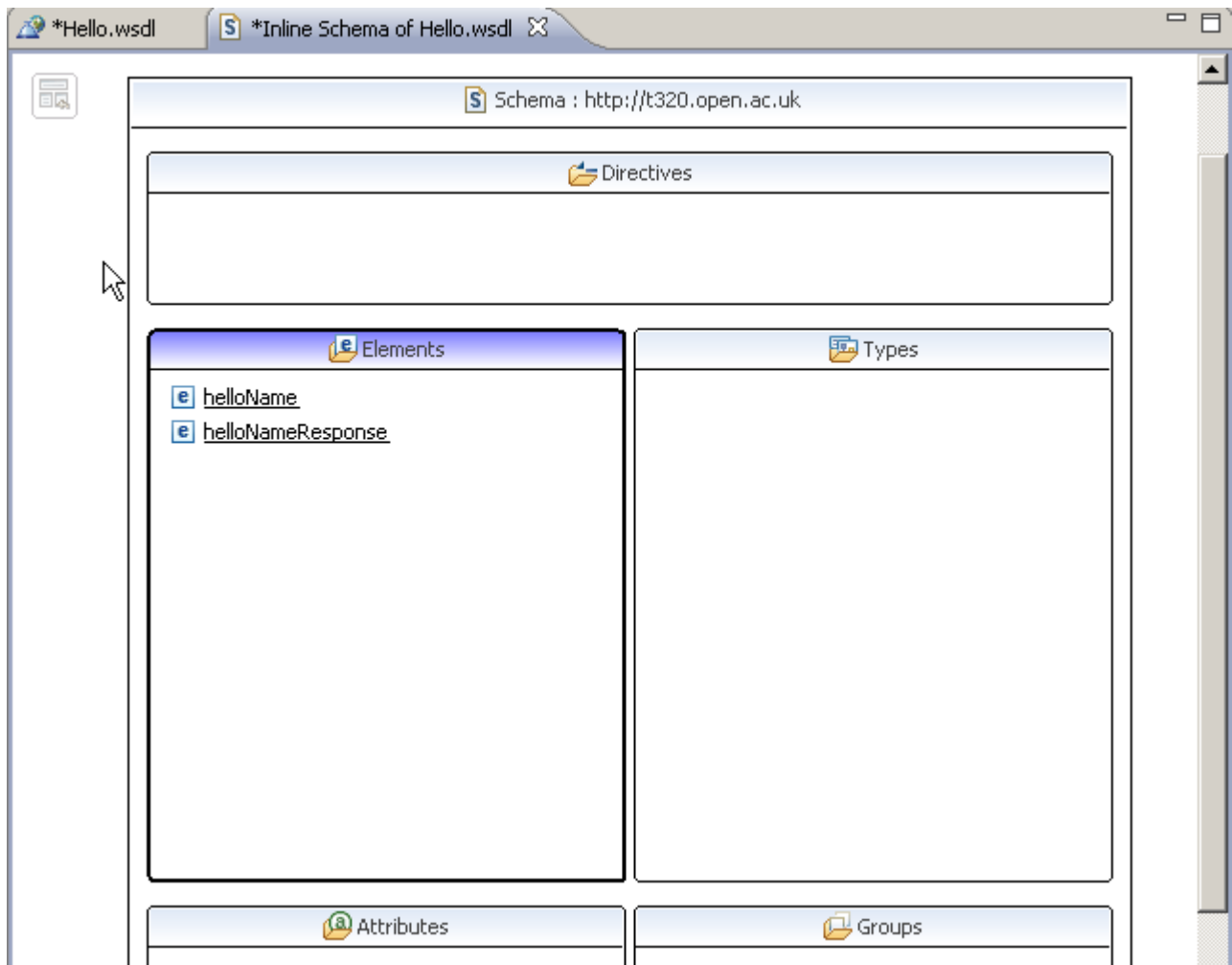
Figure 15 Index view of schema in Editor window

Summary

In this practical booklet I have illustrated some of the features of the Eclipse WSDL editor and looked in detail at how to perform a few operations, such as altering the EPR of a web service.

There is currently very little documentation covering the WSDL editor, but in general it seems possible to understand the facilities just by trying things out.

Eclipse and the WTP are a very useful tool for editing WSDL, as they provide context- and syntax-sensitive editing facilities. There are quite a few different 'views' and sub-editors, which can be confusing.

There is a range of other editing tools that have WSDL support. If you have experience of any of these, please post to the course forums any views you have on their usefulness etc in comparison to Eclipse.