

IBM Integration Bus

Pattern Authoring Lab 3

Setting parser properties
with XPath expressions

June, 2013

Hands-on lab built at product
code level version 9.0

1. Lab Objectives

In this lab, you will see how to configure the parser properties on an MQInput node. Configuring the message domain, message model, message and physical format properties on input nodes is a very common requirement in pattern authoring.

This lab uses a simple message flow with an MQ Input node. The lab will develop the pattern built in the first and second labs. The MQ Input node requires the specification of parser properties (even if defaulted). If the XMLNSC domain is selected, the message model, message and physical format parameters are not used. If the MRM domain is selected, the other parser parameters are required. This pattern enables these parameters to be specified according to selected domain.

The starting point for this demonstration is the same message flow and pattern that were created in labs 1 and 2.

This approach can be extended to many similar functions with the Integration Toolkit.

Extending the pattern to specify parser properties

1. Start this lab with the Pattern Authoring editor for the pattern “MyPattern”, as used in the previous lab session.

If you have still got the second open instance of the Integration Toolkit, close it now, and use the primary instance.

Select the message flow Transform.msgflow (you may need to open it from the Transform project).

The screenshot displays the IBM Integration Bus Pattern Authoring editor. The main workspace shows a message flow diagram with three nodes: 'MQ Input', 'Transform', and 'MQ Output'. A fourth node, 'Log queue output', is also connected to the 'Transform' node. The left-hand side features a palette with various components like Selection, Connection, Note, and several adapters (WebSphere MQ, JMS, HTTP, Web Services, SCA, WebSphere Adapters, Routing, Transformation, Construction, Database, File). Below the workspace, the 'Properties' pane is open, showing the 'MQ Input Node Properties - MQ Input'. The 'Input Message Parsing' section is expanded, revealing fields for 'Message domain', 'Message set', 'Message type', and 'Message format'. The 'Message domain' field is currently set to '<Select a domain to use when no domain is present in the message. Default is BLOB>'.

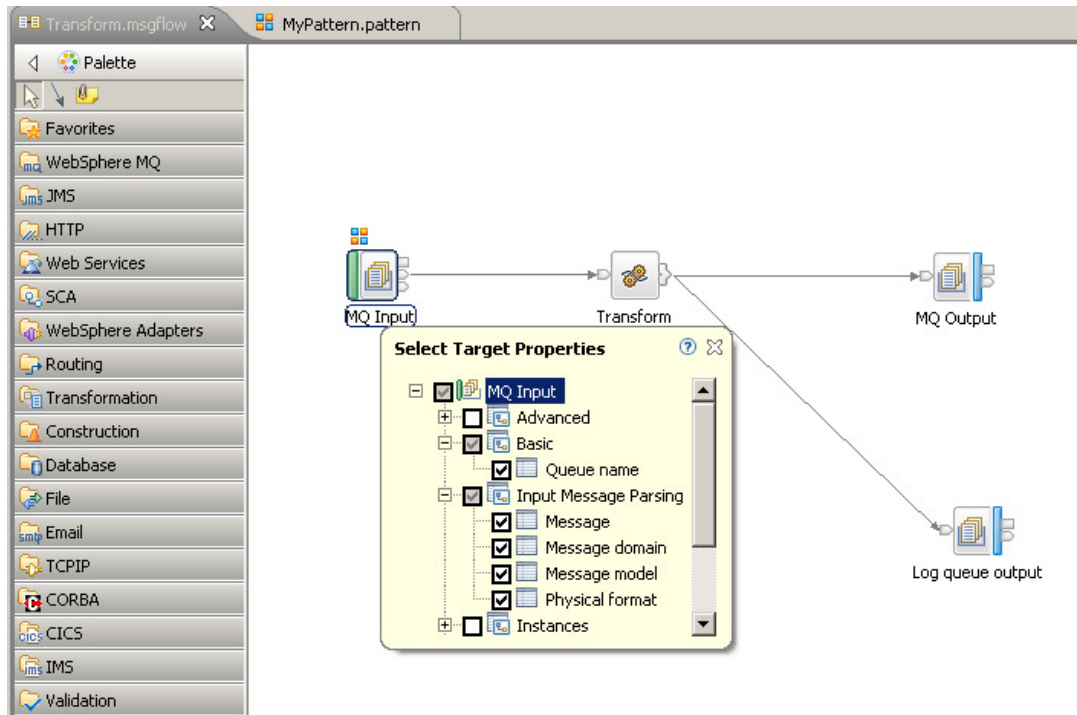
This screen shows the message parsing properties of the MQ Input node. The default message domain is BLOB. This pattern will allow the pattern user to choose between XMLNSC and MRM. Other options will be removed from the selection list.

2. Right-click the MQ Input node, and select Patterns -> Select Target Properties (or you can click on the icon directly).

Expand the MQ Input node, expand Input Message Parsing.

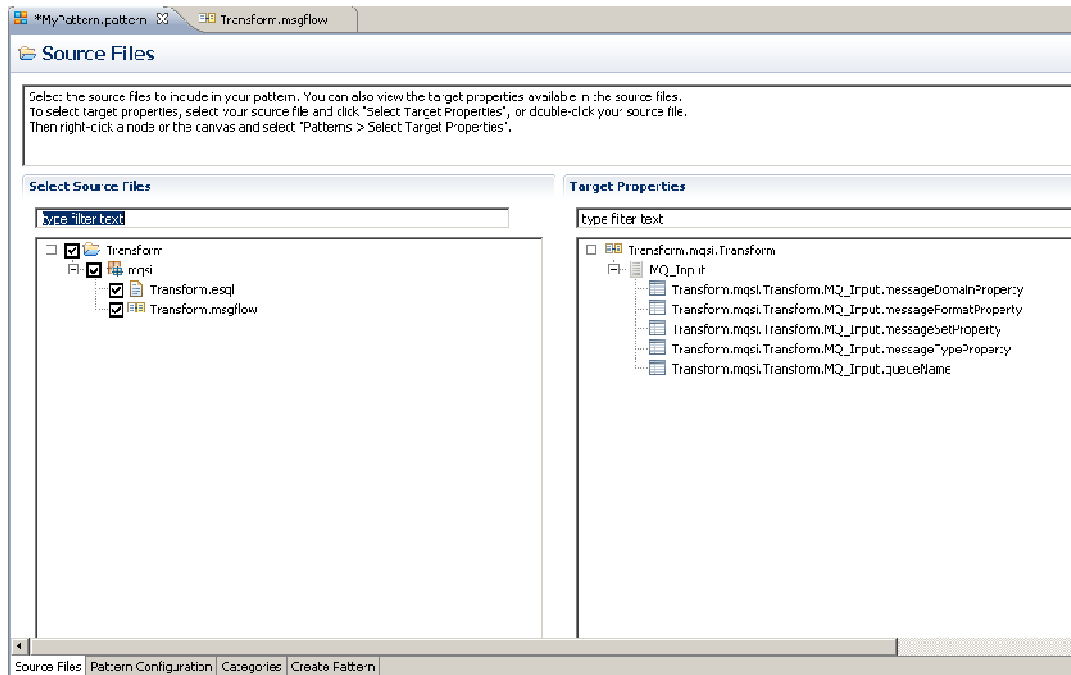
Select all the properties in the parsing group (or select the parsing group itself).

Close the Target Properties pop-up.



Save the updated message flow, and close the flow editor.

3. In the pattern editor, select the "Source Files" tab. See the new Target Properties have been added (right pane).



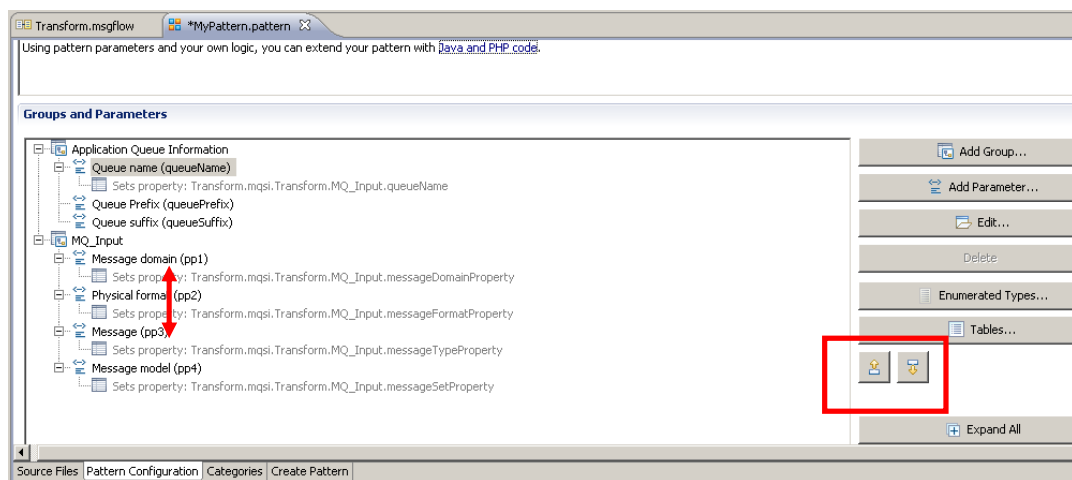
4. Switch to the Pattern Configuration tab.

The parser properties are initially not in the correct logical order. Use the Up and Down buttons (highlighted) to correct the order. (For example, highlight “Message Set”, and click the Up button).

Alternatively, you can drag and drop the Parameter names within the list.

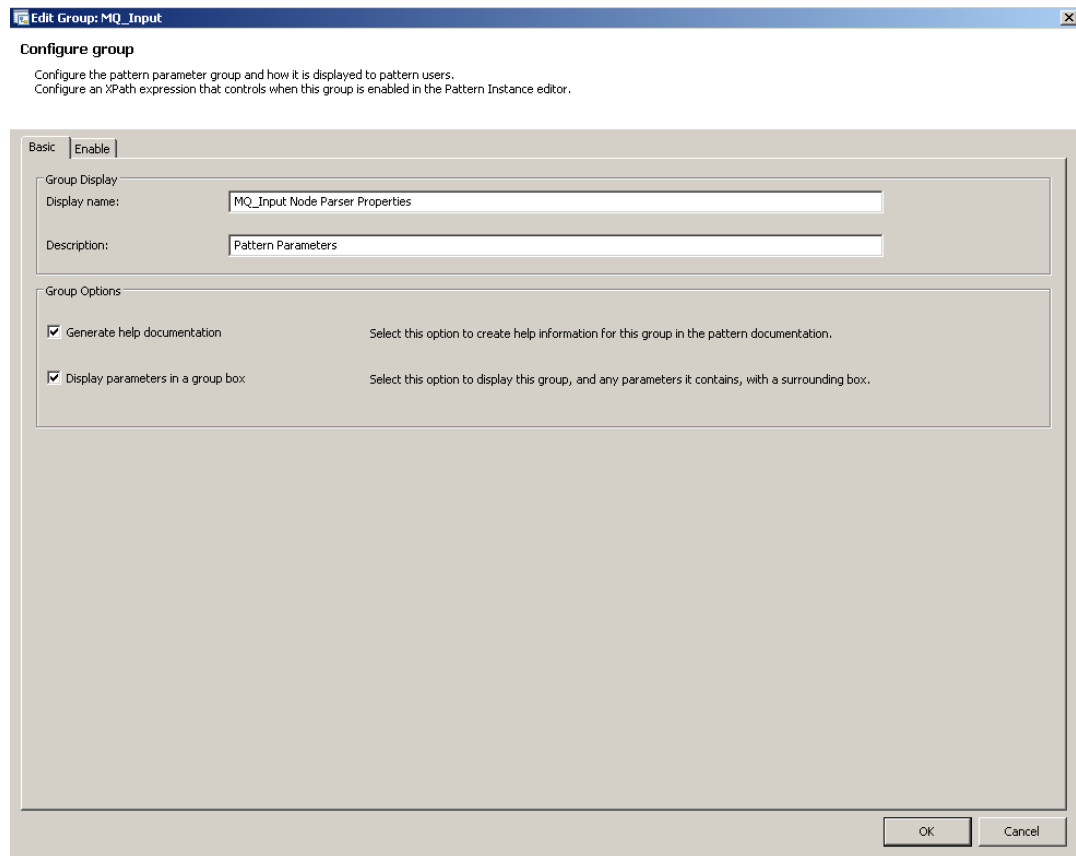
Change the order to:

1. Message Domain
2. Message Model
3. Message
4. Physical format

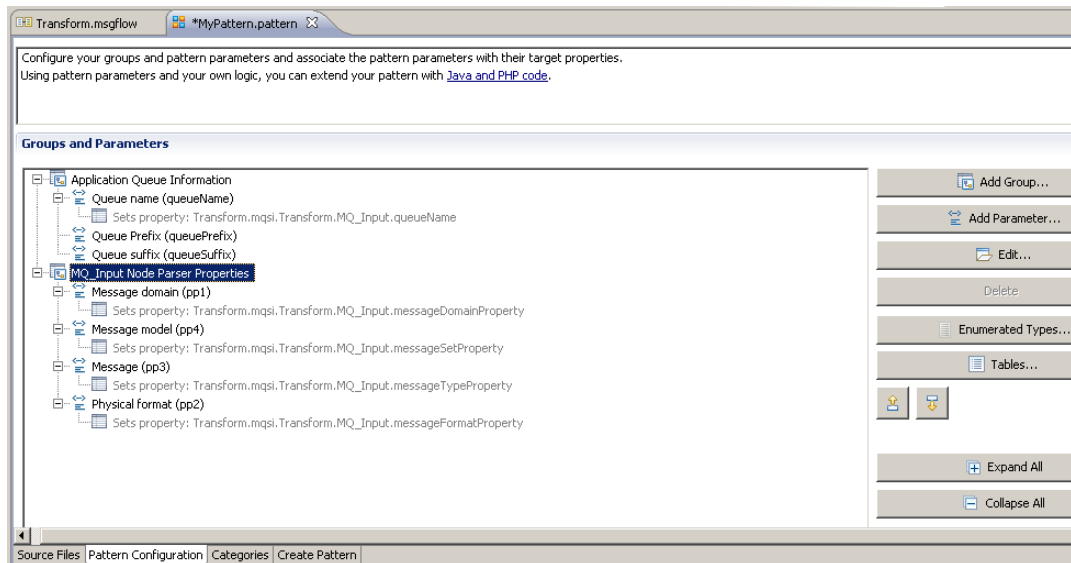


5. Finally, change the name of the parser properties group for the MQ Input node. Double-click the MQ_Input group, and change the name as shown.

Click OK.



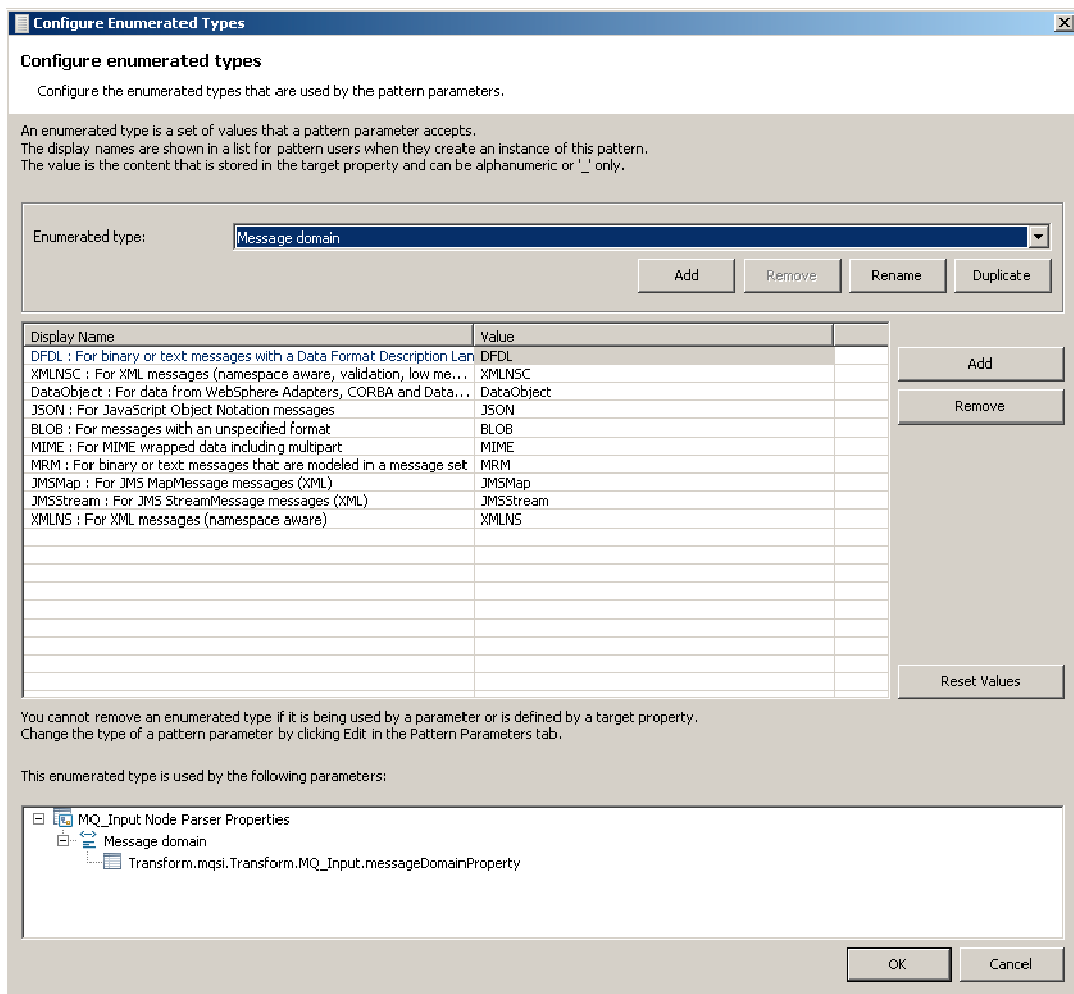
6. After making these changes, you should have something like this:



- The Message Domain is an interesting target property, because it has a pre-defined list of permissible values. This is called an enumerated type in pattern authoring. We will configure constraints on this property now, to reflect these values.

Click Message Domain, and click Enumerated Types. This will open this window.

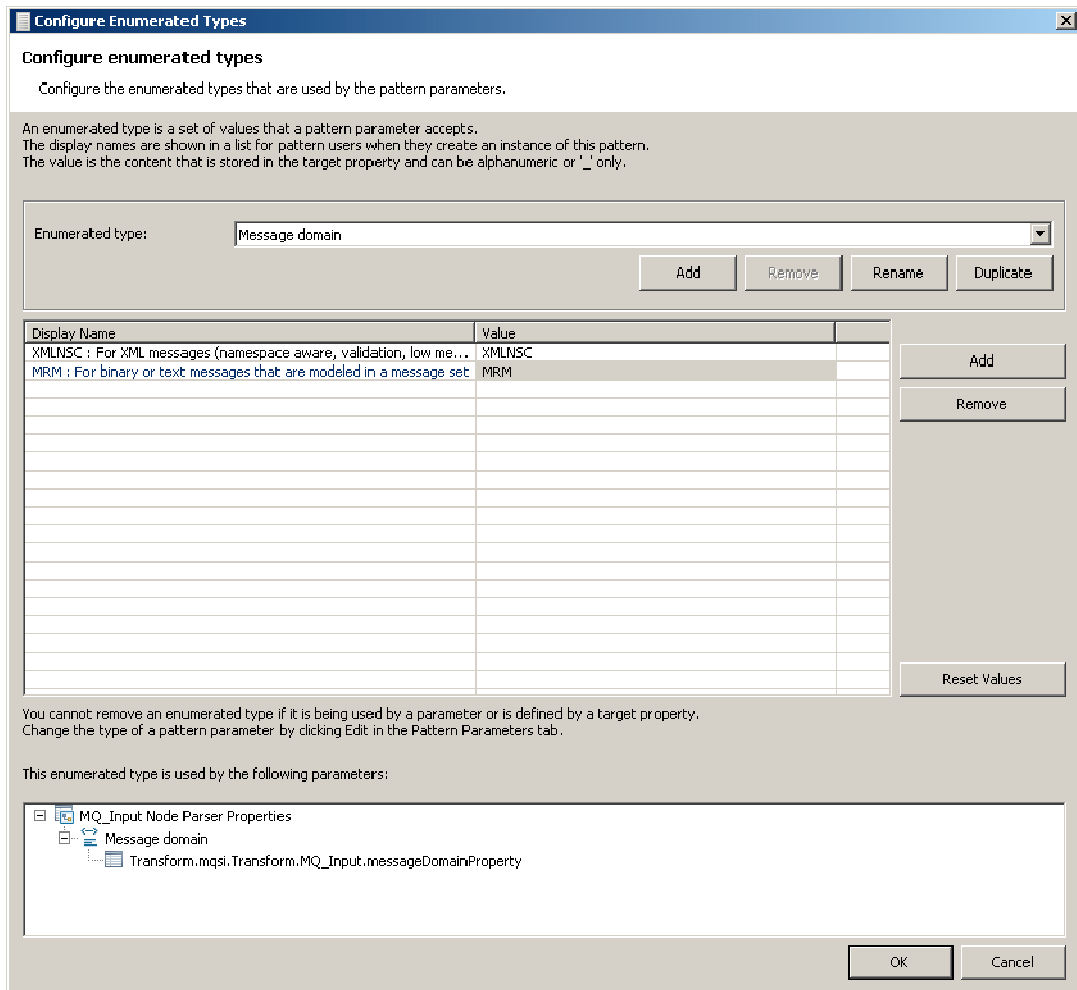
This dialogue allows you to look at and configure enumerated types. When the message domain target property was added to this pattern, the Pattern Authoring editor automatically created an enumerated type for it. The permissible values for the enumerated type are shown here. Each entry has a display name, and a value which the target property can be configured with (such as MRM and XMLNSC).



- 8. We will remove all the entries in this list except for XMLNSC and MRM. If you make a mistake, you can reset this list by clicking on the Reset Values button below. You can also see at the bottom of this dialog, that the message domain pattern parameter is using this enumerated type.

Select the unwanted lines from the list, and click Remove.

Click OK to return to the Pattern Configuration window.



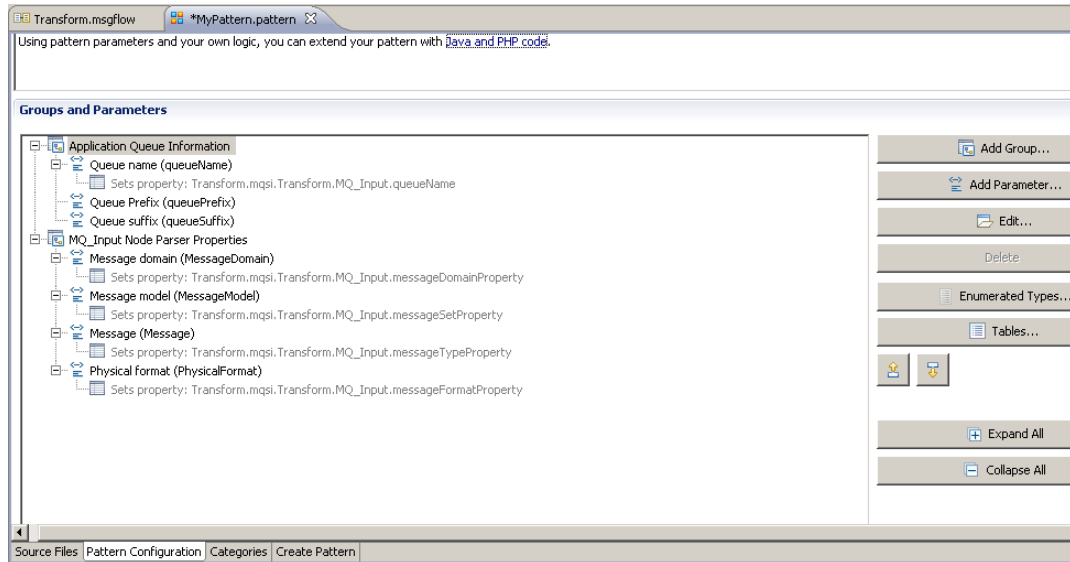
- At this point, it would be sensible to change the generated pattern parameter ID to something more useful. For the Message Domain, we will change it to MessageDomain.

To do this, highlight the Message Domain property, and click Edit. Change the Parameter ID field, and click OK.

The screenshot shows a dialog box titled "Edit Parameter: Message domain". The main heading is "Configure pattern parameter" with a sub-heading "Configure the pattern parameter and how it is displayed to pattern users." There are four tabs: "Basic", "Editor", "Transform", and "Enable", with "Basic" selected. The "Parameter Display" section has a "Display name" field with "Message domain" and a "Parameter ID" field with "MessageDomain", the latter being highlighted with a red box. The "Parameter Options" section includes a "Hide the parameter" checkbox (unchecked), a "Mandatory parameter" checkbox (checked), and a "Field prompt" field with "Enter your parameter value". The "Help Text (HTML)" section has a text area with "<p>Describe the parameter here</p>" and a "Preview parameter help" button. The preview area shows "Describe the parameter here".

10. Make similar changes to the other parser pattern parameter IDs. Set them to MessageModel, Message and PhysicalFormat.

At this point, the pattern configuration should look like this. Note that the Parameter IDs for each of the properties is shown in brackets, following the name of the property.



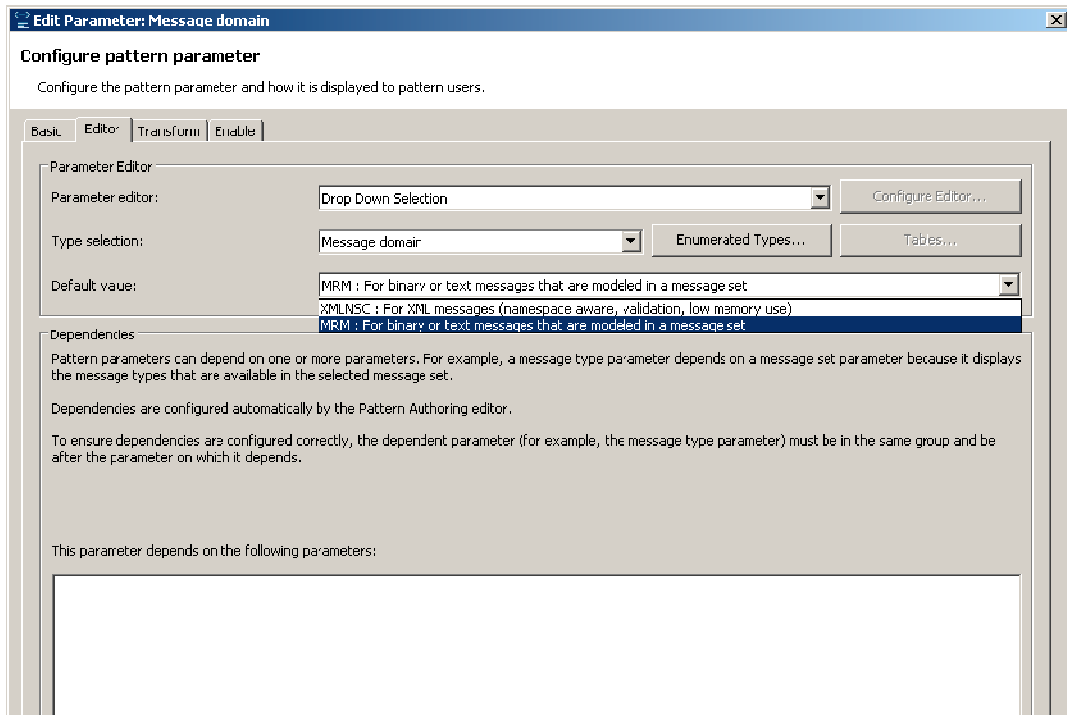
11. Now we are going to create expressions which will dynamically enable or disable parser properties, based on the selected Message Domain.

Select the Message Domain property, and click Edit.

Select the Editor tab. You will see that the “Parameter editor” has automatically been set to Drop Down Selection, and the default values are restricted to MRM and XMLNSC.

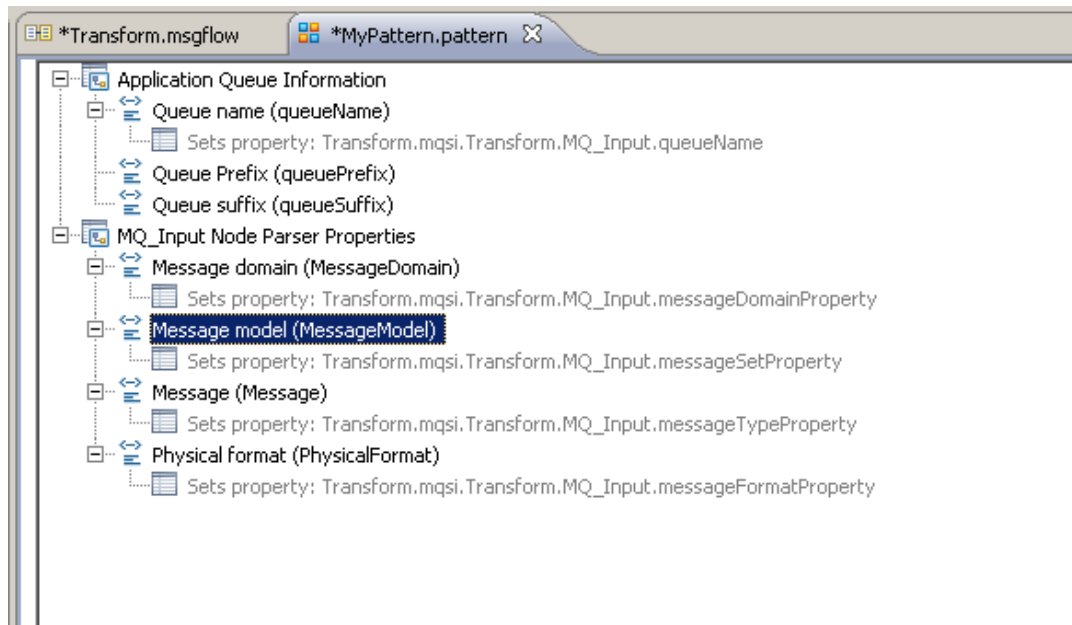
Set the default value to MRM.

Click OK.



12. If the pattern user selects XMLNSC, then you will want to disable the option of selecting the MRM parser properties (message model, message and physical format). We will do this by constructing an XPath expression for the other parser parameters.

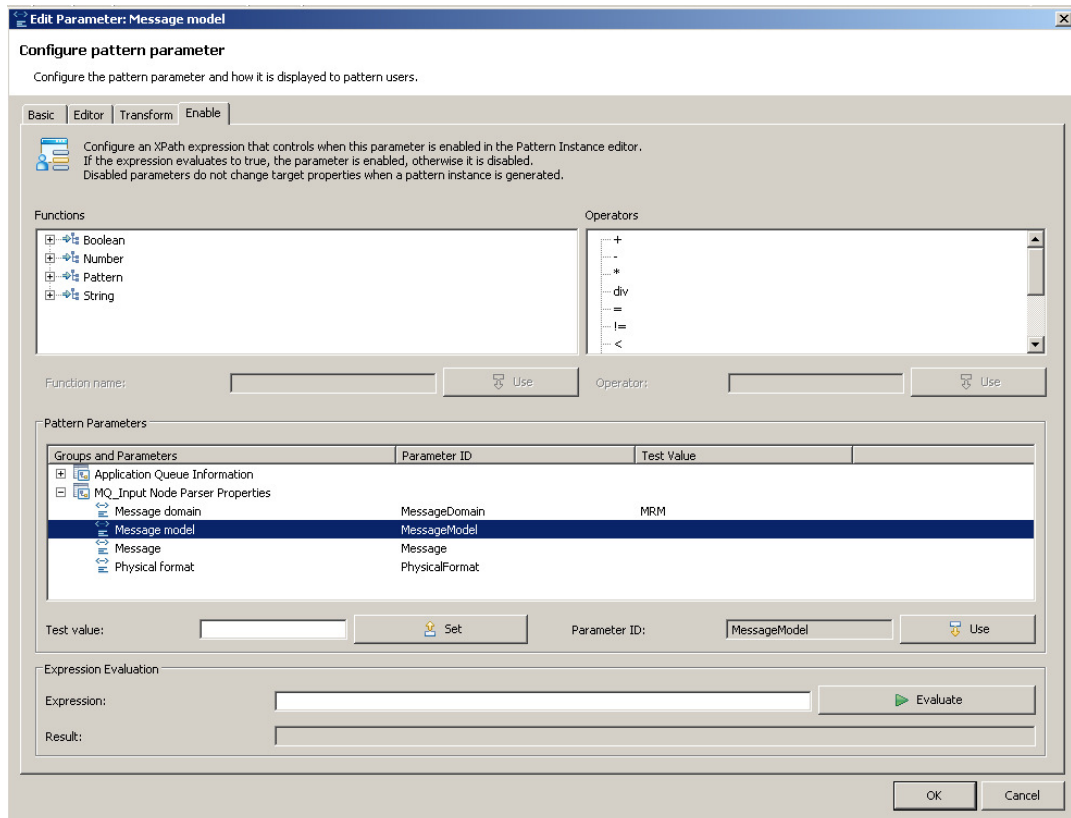
Double-click the Message Model property, to edit it..



13. Select the Enable tab.

Using the Expression field, we will construct an XPath expression. This will enable or disable the Message Set property, based on the value of the Message Domain parameter entered by the pattern user.

If the expression evaluates to “true”, then the field will be displayed to the pattern user. If it evaluates to false, it will not be displayed.



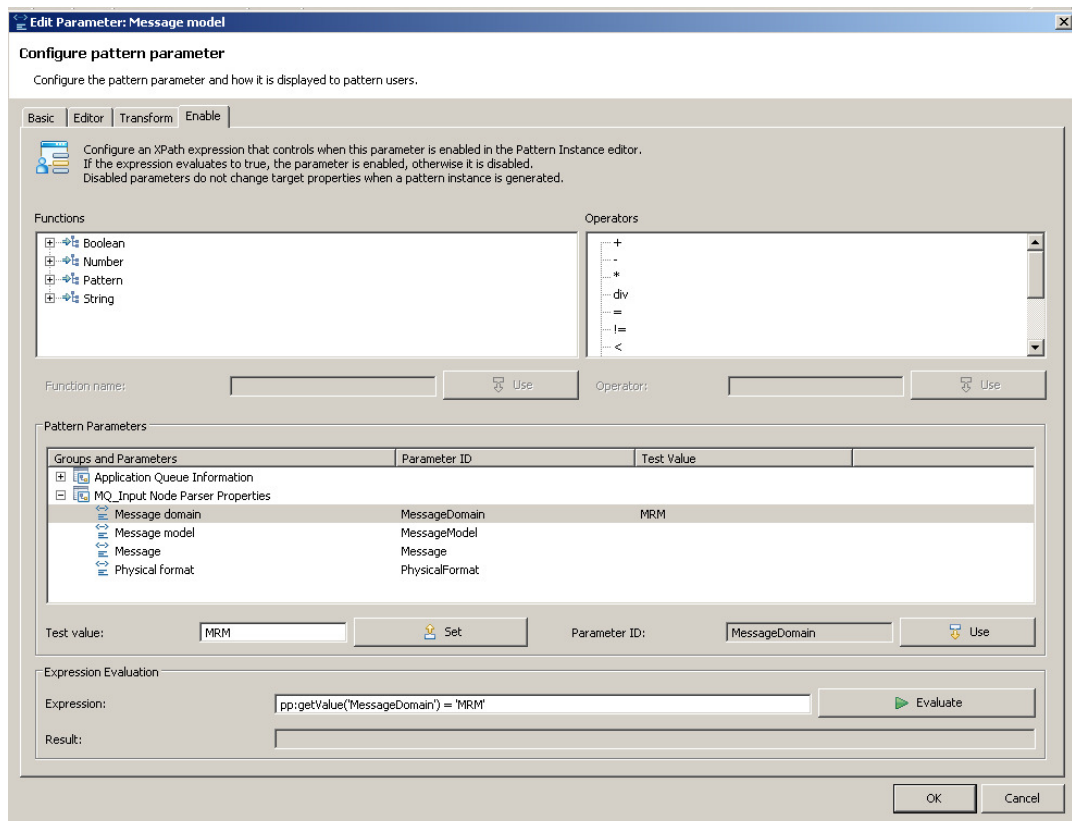
14. Double-click the Message Domain field. This will populate the Expression field with

```
pp:getValue('MessageDomain')
```

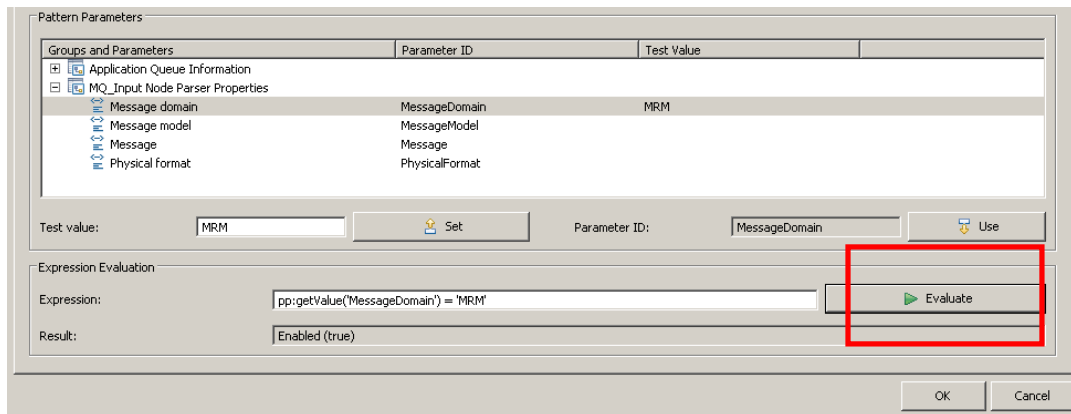
Complete the expression by typing the value to compare the expression with, as follows:

```
pp:getValue('MessageDomain') = 'MRM'
```

This will Enable or Disable the Message Set property, based on the value of the Message Domain property.

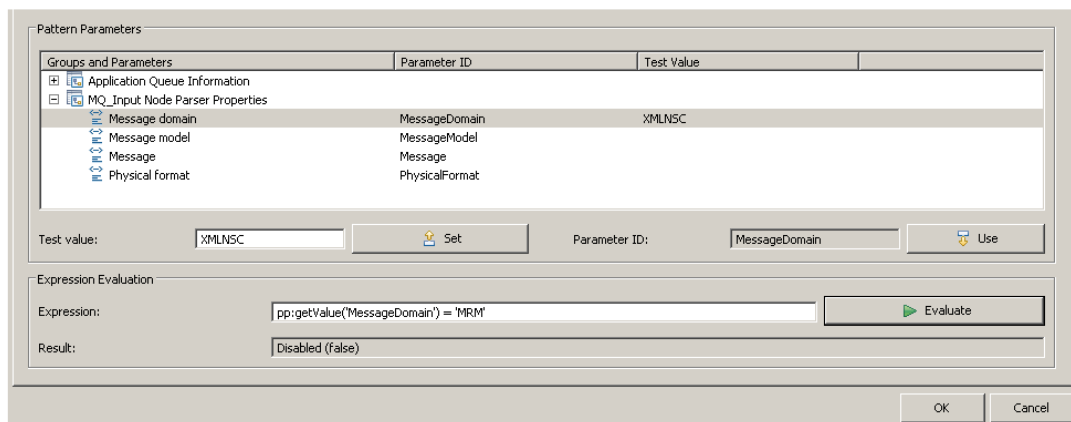


15. Check the XPath expression by using the Evaluate button. Here is the expected output.
Use the “Test value” field and the Set button to check correct operation of the expression.



16. Check to see what happens if the domain is XMLNSC. In the “Test value” field, enter XMLNSC and click Set.

Click Evaluate. The expected output is this, where the result is “Disabled(false)”. In this case, the message set field would not be shown.



Before we leave this parameter, copy the value in the Expression field to the clipboard (Ctrl-C); we will use it for the remaining parser properties.

Click OK to complete the Message Model properties.

17. Now repeat the same configuration for the Message and Physical Format properties (steps 11 to 16).

Highlight each property in turn, click Edit, select the Enable tab, and paste the contents of the clipboard into the Expression field.

Click OK.

Save the updated pattern.

18. You're done with pattern configuration. Now you need to test the updated pattern.

On the "Create Pattern" tab, click Create Pattern Plug-ins.

When this is complete, click Test Pattern.

Transform.msgflow MyPattern.pattern

Create Pattern

Test your pattern by configuring your pattern plug-in information, click "Test Pattern".

⚠ The plug-ins in your workspace may be out of date and need to be re-created.

Create Pattern Plug-ins Test Pattern...

Plug-in Information

Configure the unique identifier for your pattern plug-in.

Pattern name: MyPattern

Plug-in ID: com.betaworks.ibm.domain.MyPattern

Version: 1.0.0.0

Provider: IBM BetaWorks

Description: Plug-in created by BetaWorks Early Programs

Translation Options

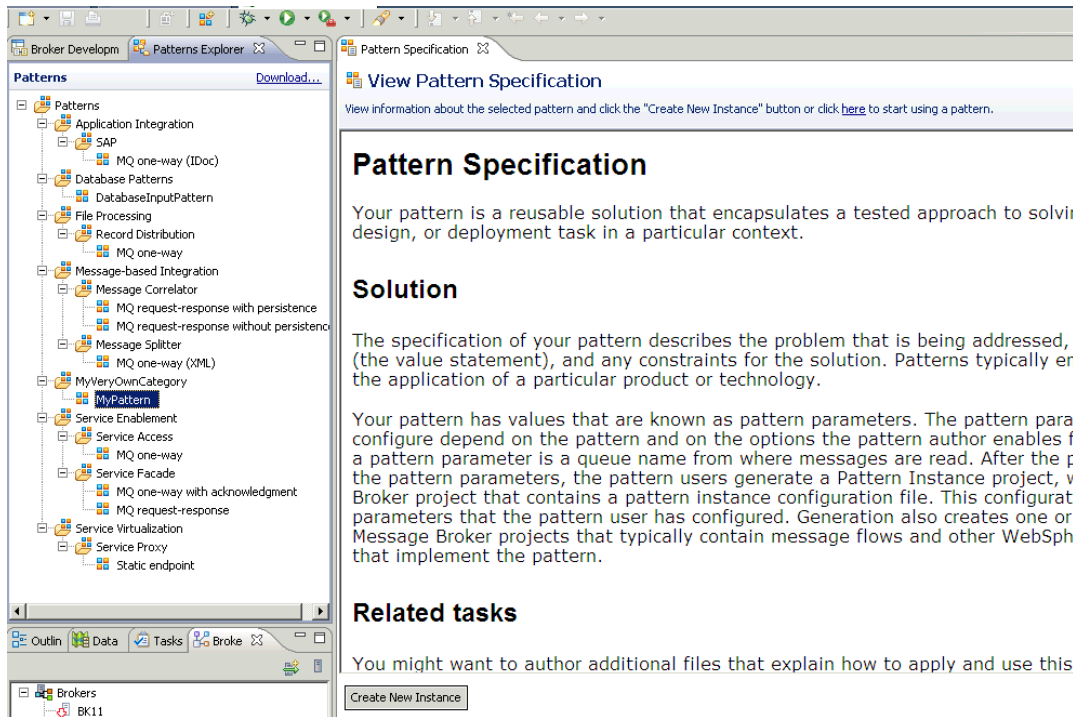
If you enable this option, the Pattern Authoring editor creates two additional NLS plug-ins. These plug-ins are used to create single language patterns. If you are creating a single language pattern, do not select this check box.

Create translation plug-ins (*.nl1 and *.doc.nl1)

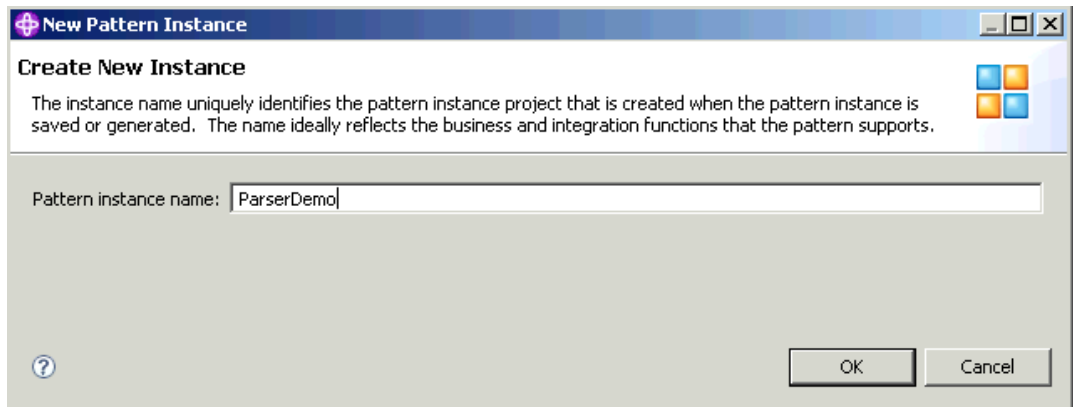
Source Files Pattern Configuration Categories Create Pattern

19. Accept the location of the second workspace.

- If not automatically selected, click on Patterns Explorer, and click MyPattern.



- Click "Create New Instance", provide a name, and click OK.



22. Expand the MQ Input Node Parser Properties group.

The default value for Message Domain is MRM. In this case, the Message model, message and physical format parameters must be provided.

For this pattern to be generated, an appropriate message set must be present in the developers workspace, so this is the limit of the part of the task.

Pattern Specification ParserDemo - Pattern Configuration

Configure Pattern Parameters

Provide values for pattern parameters. Click the "Generate" button or click [here](#) to generate a pattern instance.

✘ The pattern parameter "Message model" is mandatory but a value is not set.

Pattern Parameters

- Application Queue Information ✓
- MQ_Input Node Parser Properties ✘**
 - Pattern Parameters
 - Message domain *
 - Message model *
 - Message *
 - Physical format *

Generate

Specification Configuration

23. On the other hand, if the Message Domain is set to 'XMLNSC', then the remaining parser properties are greyed out, and will not be specified.

In this case, you can proceed to generate a new pattern instance. Use the same approach as in the earlier labs, and observe the resulting MQ Input node properties in the generated message flow.

The screenshot shows the 'Configure Pattern Parameters' dialog box. The title bar includes 'Pattern Specification' and '*ParserDemo - Pattern Configuration'. The main heading is 'Configure Pattern Parameters'. Below the heading, there is a message: 'Provide values for pattern parameters. Click the "Generate" button or click [here](#) to generate a pattern instance.' An information icon (i) is followed by the text: 'Pattern parameters are ready. Click the "Generate" button to generate a pattern instance.'

The 'Pattern Parameters' section is expanded to show 'Application Queue Information' (checked) and 'MQ_Input Node Parser Properties' (checked). Under 'MQ_Input Node Parser Properties', the following fields are visible:

- Message domain *: XMLNSC : For XML messages (namespace aware, validation, low memory use)
- Message model *: <Enter your parameter value>
- Message *: <Enter your parameter value>
- Physical format *: <Enter your parameter value>

A 'Generate' button is located at the bottom left of the dialog. At the bottom, there are tabs for 'Specification' and 'Configuration'.

This concludes the Pattern Authoring Parser Properties lab.