

βetaW<mark>≎rk</mark>s

IBM Integration Bus

Pattern Authoring Lab 5

User Customisation

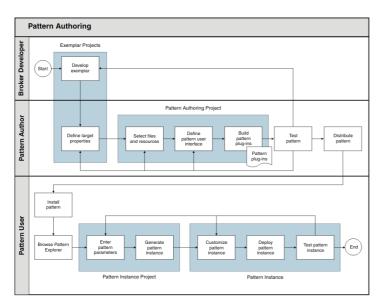
June, 2013

Hands-on lab built at product code level Version 9.0.0.0

1.	INTRODUCTION - USER CUSTOMISATION	3
2.	CREATE THE MESSAGE FLOWS	4
3.	CREATE THE PATTERN	. 12
4.	INSTANTIATE THE PATTERN	. 20
5.	CUSTOMISE THE MESSAGE FLOW	. 22
6.	REGENERATE THE PATTERN INSTANCE	. 23
7.	MODIFYING THE PATTERN CONFIGURATION	. 25
8.	INSTANTIATE THE MODIFIED PATTERN	. 30
9.	CUSTOMISE THE MESSAGE FLOW AGAIN	. 32
10.	REGENERATE THE PATTERN INSTANCE AGAIN	. 33

1. Introduction - User Customisation

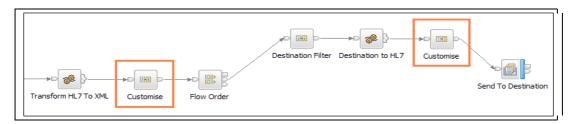
When it comes to user-defined patterns authoring, the usual workflow is the following:



It is the "Pattern Author" who defines the patterns, and a "Pattern User" who instantiates them. The pattern user usually needs to introduce some modifications to the Pattern to adapt it to the scenario.

To allow that, a new feature was introduced in Message Broker version 8 to enable the Pattern User to customise the message flows without overwriting the changes every time he re-generates the Pattern instance.

To be able to do that, the Pattern Author must include customisation subflows in the main message flow, which will be held in a separate project or Library. This library, which will contain all the customisation subflows, will have the new feature enabled, so that when the Pattern User re-generates a pattern instance, the subflows are not overwritten.



In this lab you will explore this by first creating a pattern as usual, without enabling the new feature, and then secondly with it enabled, to test its effects.

2. Create the Message Flows

1. If not already started, start the Integration Toolkit by clicking its icon in the quick launch toolbar or on the desktop. Accept the default workspace (C:\workspace\IBWorkshop).

You are going to import the same base message flow you used in the earlier pattern authoring labs.

2. Import the project interchange file by clicking on File->Import.

Select Other-> Project Interchange, and click Next.

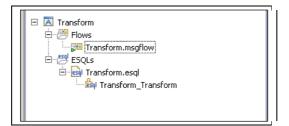
D Import	
Select Import a project and its dependent projects from a Zip file.	Ľ
Select an import source:	
General CVS EJB Modeling Plug-in Development Run/Debug C Run/Debug Web Web Web Web Web C Web Project Interchange	
Contract Sector Sect	Cancel

3. Click on the Browse button and navigate to "C:\student\PatternAuth\resources" and select the PatternsAuthBasic_ExemplarV8.zip.

🜔 Import Project In	nterchange Contents	
Import Projects Import Projects from a	a zip file.	<u></u>
From zip file: Project location root:	C:\student\PatternAuth\resources\PatternAuthBasic C:\student\workspace	Browse Browse
🗹 🗁 Transform		
Select All Deselect	t All Select Referenced	Capcel
(\mathfrak{Y})	< Back Next > Finish	Cancel

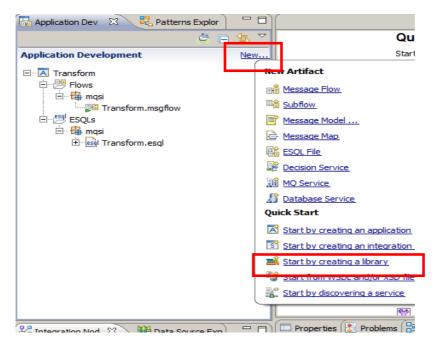
Make sure the Transform project is selected, and click Finish.

4. Your projects view should look like this. Note that the PI file has been imported as an Integration Bus Application.

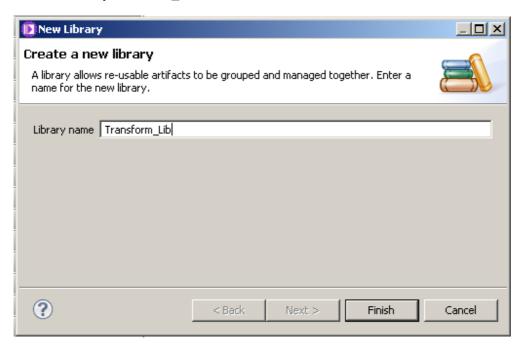


5. Create a subflow to be used as a customisation node in the main message flow. This new subflow will be stored in a Library, so first we first need to create a new Library.

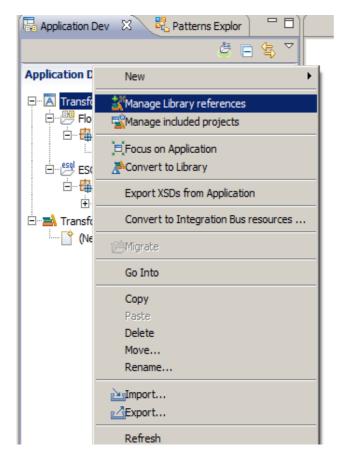
In the navigator, click New, then click "Start by creating a library".



6. Name the library Transform_Lib and click Finish.



7. Right-click the Transform application, and select Manage Library References.



8. Select Transform_Lib and click OK.

Manage Library references	×			
Check the libraries to be referenced. Any other libraries referenced from the checked library will also be included. Leaving a library unchecked will prevent the libraries from being included.				
✓ ➡ Transform_Lib				
, The following libraries will also be included as they are referenced by the checked libraries.				
OK Cancel				

9. In the new library, Transform_Lib, click (New...) to create a new subflow. On the dialogue box, click Subflow.

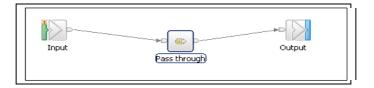
🖥 Application Dev	🛛 🧏 Patterns Explor	
	<i>ë</i> 🖻	\$₽ ~
Application Develo	opment	New
ESQLs		
	New Artifact	
	Create a new artifact.	
	Message Flow	
	^国 畲 <u>Subflow</u>	
	Message Model	
	Message Map	
	ESQL File	
	Decision Service	
9.2	MQ Service	
🖧 Integration Nod	🕼 Database Service	Ľ

10. Name the subflow TransformSub1 and click Finish.

🜔 New Subflow	v		
Create a nev Select a contai	w subflow ner for the new subflow		8-8
Container: Subflow name:	Transform_Lib		New
Flow organiza	tion It broker schema		
Schema; (de	fault broker schema)		
?		Finish	Cancel

11. When the flow editor opens, it will already have placed an Input and Output node into the flow.

Place a passthrough node between these and wire them as shown.



12. Press Ctrl+S or File->Save to save the message flow.

Close the message flow editor.

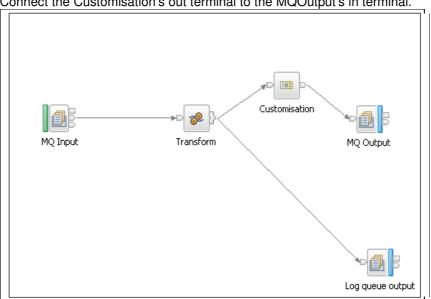
13. Now you are going to include the subflow in the Transform message flow.

Open the Transform message flow by double-clicking on its name.

Drag and drop the subflow on to the flow editor. Rename the Subflow node to "Customisation".

🔚 Application Dev 🛛 🧏 Patterns Ex	(plor)	😬 Transform.msgflow 😫			- 8
	5 🔁 🗧	👌 😳 Palette	2		
Application Development	New	<u> </u>			
⊡- A Transform		🙀 Favorites			
🕂 🕮 Flows		🖳 WebSphere MQ			(Customisation)
🖻 📲 mqsi		Gins JMS			Castomedaday
Fransform.msgflow □ 1991 ESQLs		💭 HTTP			
		🧟 Web Services			
		😨 SCA	T E B		
E References		WebSphere Adapters	MQ Input	Transform	MQ Output
⊡ <u>=</u> A Transform_Lib ⊡ ⊡ Subflows		Routing			
TransformSub1.subflow		💫 .NET			
-		Transformation			
		Construction			
		Database			
		🥪 File			► 0 8
		🔜 Email			Log queue output
		CPIP			
		Graph User Defined Properties			
	-)	Properties R Problems	🗄 Outline 🧔 Tasks 🔲 Deployment Log	Console	

14. Connect the out terminal of the Transform node to the Customisation's in terminal.



Connect the Customisation's out terminal to the MQOutput's in terminal.

15. Save the message flow by pressing Ctrl+S or File->Save.

3. Create the Pattern

1. Create a new Pattern, by clicking File-> New -> Pattern Authoring Project or clicking on the New icon:

ſ	C1 • 😫 🖬 • 🔛 🕼 📤	Ð	: 😭
Ē	Application		
	🛋 Library		
	👼 Message Broker Project		
ľ	🔡 Pattern Authoring Project		
1	📸 User-defined Node Project		
	🃸 Project		
	📸 Message Flow		
	🛺 Message Map		
	🎼 Message Flow ESQL File		
	📸 Adapter Connection		
	📋 Database Definition		
	👔 BAR file		
F	🞼 Message Broker Test Client		
	💣 User-defined Node		
	📬 Example		
	📬 Other	Ctrl-	FΝ

2. Enter "CustomisationPattern" as the pattern name, and "CustomisationPatternProject" as the project name.

Click Next.

D New Patter	n Authoring Project	
	Pattern Authoring Project tes a new Pattern Authoring project.	
Pattern name:	CustomisationPattern	
Project name:	CustomisationPatternProject t location	
Location: C:\s	student\workspace\CustomisationPatternProject	Browse
?	< <u>B</u> ack <u>N</u> ext > <u>F</u> inish	Cancel

3. Select the two projects and click Finish.

🜔 New Pattern Authoring Project	<u>- 0 ×</u>
Create a New Pattern Authoring Project	
Specify dependencies on other projects.	
Referenced projects:	
Image: Transform_Lib	
	Cancel

4. The Pattern Authoring Editor will open with the newly created CustomisationPattern.

Select the Transform project and click on the Edit button.

🕫 Transform.msgflow 🧧 🔡 CustomisationPa	🗉 Transform.msgflow 🗧 🚼 CustomisationPattern.pattern 🛛						
∋ Source Files							
Select the source files to include in your pattern. You can also view the target properties available in the source files. To select target properties, select your source file and click "Select Target Properties", or double-click your source file. Then right-click a node or the canvas and select "Patterns > Select Target Properties".							
Select Target Properties	🕞 Change Project References	Edit					
Select Source Files		Target Properties					
type filter text		type filter text					
Image: Second system Image: Second system							

5. Make sure the "Pattern users will modify this project" is unchecked. Click Cancel.

🖨 Edit Project: Transform					×
Configure project					
Configure the project and how it is configure Use an XPath expression to control when thi	ed when a pattern instance is project is generated in a	e is generated. pattern instance.			
Basic Create					
Project Options					
Pattern users will modify this project		If this option is selected, files in instance is re-generated.	this project will not be a	overwritten when a p	attern
Prefix the project name with the patte	rn instance name	Select this option and the project name when it is generated.	t name will be prefixed	with the pattern inst	ance
Select a pattern parameter for the project	t name when the pattern i	nstance is generated:			
Pattern parameter name:			Clear	Select	
				ОК	Cancel

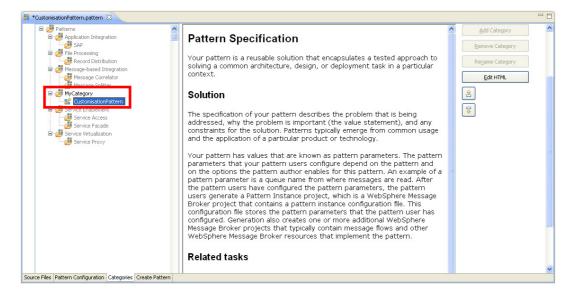
Repeat this step with the Transform_Lib project.

6. In the Pattern Authoring Editor, click on the Categories tab.

Then click on the "Add Category" button. Enter "MyCategory" as the category name and click OK.

A popup will appear asking you to save your pattern. Click Yes.

7. Now drag the CustomisationPattern and drop it under the newly created MyCategory category.

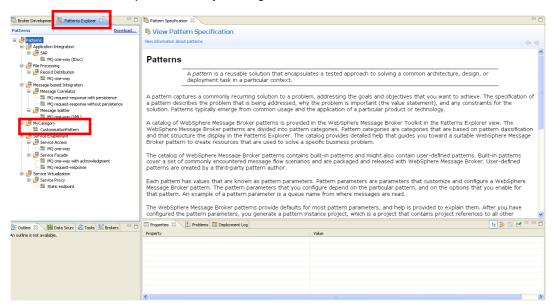


8. In to the "Create Pattern" tab and click on the "Test Pattern" button. This will start a new IBM Integration Toolkit instance.

🖽 Transform.msgflow	*CustomisationPattern.pattern 🛛
进 Create Pattern	
Test your pattern by configur	ing your pattern plug-in information, click "Test Pattern".
🛃 Create Pattern	Plug-ins 🚺 Test Pattern 🎋 Debug Pattern
Plug-in Information	
😰 Configure the unique id	dentifier for your pattern plug-in.
Pattern name:	CustomisationPattern
Plug-in ID:	com.your.company.domain.CustomisationPatternProject
Version:	1.0.0.0
Provider:	Your Company Name
Description:	Plug-in created by the Pattern Authoring editor
Translation Options	
	ie Pattern Authoring editor creates two additional NLS plug-ins. These plug-ins are set up so that you can drop in trai language pattern, do not select this check box.
Create translation plug-	ins (*.nl1 and *.doc.nl1)
Pattern Distribution	
	anthere also include the state of a three and include the skilling "County Dathere Angling" and the stitution to the
	pattern plug-ins, package them into a pattern archive by clicking "Create Pattern Archive" and then distribute the pa
	attern user must launch the pattern archive file in a web browser, launch the pattern archive file in a file explorer, or
For more information about	packaging and distributing user-defined patterns, click <u>here</u> .
Source Files Pattern Configurat	ion Categories Create Pattern

9. In the workspace selection window, accept the default workspace and click OK.

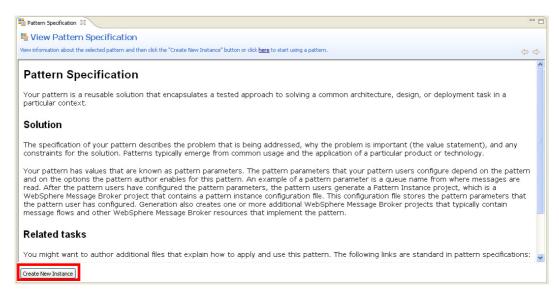
10. Go to the Patterns Explorer view, by clicking on the tab.



Note that the created pattern "CustomisationPattern" is under "MyCategory".

4. Instantiate the Pattern

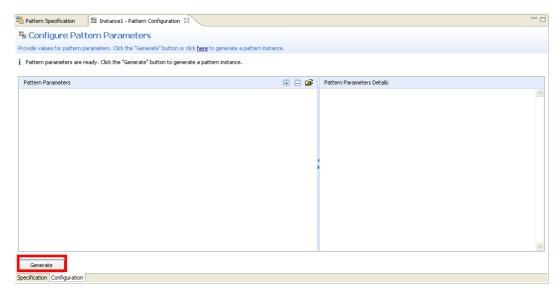
1. Click on "CustomisationPattern" and when its specification opens, click on the "Create New Instance" button.



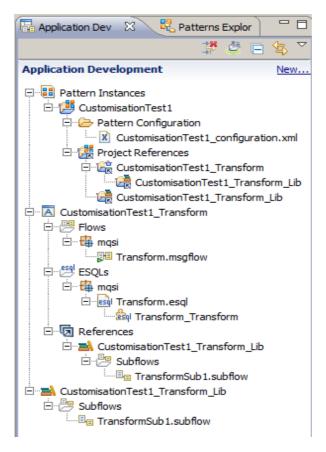
2. Enter "CustomisationTest1" as the name for the instance and click OK.

[New Pattern Instance	
Create New Instance The instance name uniquely identifies the pattern instance project that is created when the pattern instance is saved or generated. The name ideally reflects the business and integration functions that the pattern supports.	
Pattern instance name: CustomisationTest1	
(?) ОК	Cancel

3. Now click on the Generate button. Note that no pattern parameters will be shown because we haven't defined any in the template.



This will create a new Application and Library, along with the associated Library Reference. They will be prefixed CustomisationTest1, since our pattern chose to keep these prefixes.



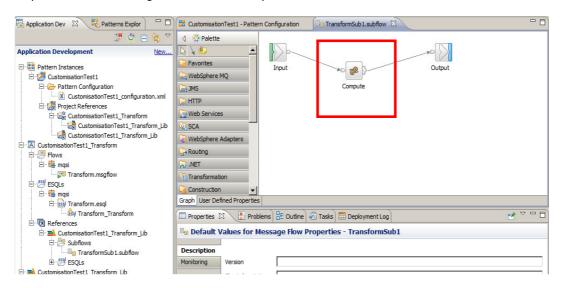
Pattern Authoring – User Customisation Provided by IBM BetaWorks

5. Customise the Message Flow

1. Next you are going to make a change to the Subflow, and see what happens when you regenerate the pattern instance.

In the second (test) workspace, open the Subflow message flow TransformSub1 by doubleclicking on its name in the Library (or direct from the Library Reference).

Replace the Passthrough node with a Compute node as shown.



Connect the three nodes:

Input (out) -> Compute (in) Compute (out) -> Output (in)

2. Double-click on the Compute node to generate the default ESQL file.

Close it by clicking on the "X".

3. Save the subflow by pressing Ctrl+S or File->Save.

Close the Subflow message flow.

6. Regenerate the Pattern Instance

1. Return to the Pattern Instance configuration by clicking on the "CustomisationTest1 - Pattern Configuration" tab, or double-clicking on the "CustomisationTest1_configuration.xml" file under CustomisationTest1 -> Pattern Configuration.

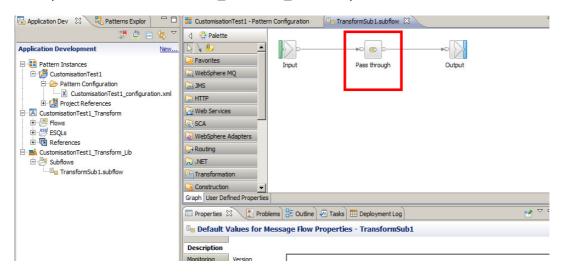
		Instance1 - Pattern Configuration 🛛			- 8
🖫 🛎 🖻 😫	🞽 🐁 Configure Patter	meaneters			
Working set: Instance1		meters. Click the "Generate" button or click <u>her</u>	e to generate a pattern instance.		
Applications and Libraries Quick Starts	i Pattern workspace resources	s are generated successfully!			
🖶 🏥 Instance1					
Generation Generation Generation	Pattern Parameters		E E 🗳	Pattern Parameters Details	
Instance1_configuration.xml Instance1_manscomeq.pp					9
Instance1_TransformUb					
Projects New					
🗉 🥵 Instance1_Transform					
😑 💯 Flows					
🖻 🛱 masi					
🖻 👹 ESQLs					
Instance1_TransformSubflow Instance1_TransformSubflow					
(default broker schema) Subflow.msgflow					~
E-OLS					
	Generate				
	Specification Configuration				

2. Click on the Generate button.

A popup window will appear asking if you want to overwrite the existing instance. Click OK.

🜔 Patt	ern Instance Generation	:		
♪	The project CustomisationTest1_Transform_Lib, CustomisationTest1_Transform cannot be generated because it exists in you workspace. Click OK to delete the project.			
	OK Cancel			

3. Now open the "CustomisationTest1_Transform_Lib" and open the TransformSub1 subflow.



Note that the changes you've made (replacing the passthrough node with a compute node) are gone. The re-generation of the pattern's instance has overwritten your customisation.

7. Modifying the Pattern Configuration

1. Using a new feature in Pattern Authoring, available since Message Broker Version 8, you are going to overcome this issue.

Close the current Integration Bus Pattern Instance Test Toolkit and go back to the original Toolkit instance.

You should be in the Pattern Authoring Editor with the CustomisationPattern pattern open.

Click on the "Source Files" tab.

CustomisationPatte	ern.pattern 🕱		
进 Create Pat	ttern		^
Test your pattern by	y configuring your pattern plug-in information, click "Test Pattern".		
Plug-in Informati	ion e unique identifier for your pattern plug-in.		
Pattern name: Plug-in ID: Version: Provider: Description: Translation Optio If you enable this	CustomisationPattern com.your.company.domain.CustomisationPatternProject 1.0.0.0 Your Company Name Plug-in created by the Pattern Authoring editor	Create Pattern Plug-ins Test Pattern File Debug Pattern Nat you can drop in translated	
Pattern Distribut	tion plug-ins (*.nlt and *.doc.nlt) ion reated and tested your pattern plug-ins, package them into a pattern archive by clicking "Create and then distribute the pattern archive file to your pattern users.	uter and the street of the str	
To install the patte	and then bachduce the pattern and here the to your pattern takes. erry, the pattern user must launch the pattern archive file in a web browser, launch the pattern archive res. delt. The pattern states are the pattern of the pattern archive file in a web browser, launch the pattern archive Configuration [Categories] Create Pattern		

2. In the "Source Files" section select the TransformLib project. Note that hovering over the Transform_Lib will show a pop-up with brief details of the configuration.

📧 *Transform.msgflow 🛛 🔡 CustomisationPattern.pattern 🛛	
😂 Source Files	
Select the source files to include in your pattern. You can also view the target properties availa To select target properties, select your source file and click "Select Target Properties", or doubl Then right-click a node or the canvas and select "Patterns > Select Target Properties".	
Select Source Files	Target Properties
type filter text	type filter text
Image: Transform Image: Transform.esql Image: Transform.msgflow Image: Transform_Lib Pattern users will modify this project: No Prefix the project name with the pattern instance name: Yes To change these settings, click on your project name and click "Edit", or d	suble-click your project name.
Source Files Pattern Configuration Categories Create Pattern	

Click on the Edit button.

CIII Transform.msgflow				
😂 Source Files				
Select the source files to include in your pattern. You can also view the target properties available in the source files. To select target properties, select your source file and click "Select Target Properties", or double-click your source file. Then right-click a node or the canvas and select "Patterns > Select Target Properties".				
🔲 Select Target Properties 🔁 Change Project References	🔁 Edit			
Select Source Files	Target Properties			
type filter text	type filter text			
Image: Construction of the second				

3. Check the "Pattern users will modify this project" and click OK.

🚖 Edit Project: Transform_Lib	X
Configure project	
Configure the project and how it is configured when a pattern instance is generated Use an XPath expression to control when this project is generated in a pattern insta	j. Ince.
Basic Create	
C Project Options	
Pattern users will modify this project	If this option is selected, files in this project will not be overwritten when a pattern instance is re-generated.
☑ Prefix the project name with the pattern instance name	Select this option and the project name will be prefixed with the pattern instance name when it is generated.
Select a pattern parameter for the project name when the pattern instance is gen	nerated:
Pattern parameter name:	Gear Select

Save the CustomisationPattern.pattern.

4. Click on the "Create Pattern" tab.

Click "Create Pattern Plug-ins".

When complete, click on the "Test Pattern" button.

📧 Transform.msgflow	🖁 CustomisationPattern.pattern 🛛
⁴ Create Patterr	۱
· · · · ·	
Test your pattern by config	guring your pattern plug-in information, click "Test Pattern".
🔥 The plug-ins in your w	vorkspace may be out of date and need to be re-created.
🔗 Create Patter	rn Plug-ins 🚺 💽 Test Pattern
Plug-in Information	
😰 Configure the uniqu	e identifier for your pattern plug-in.
Pattern name:	CustomisationPattern
Plug-in ID:	com.your.company.domain.CustomisationPatternProject
Version:	1.0.0.0
Provider:	Your Company Name
Description:	Plug-in created by the Pattern Authoring editor
Translation Options	
	, the Pattern Authoring editor creates two additional NLS plug-ins. These plug-ins are set up so that you can drop gle language pattern, do not select this check box.
- ,	····· 9-9- F-···· , -····
Create translation plu	ug-ins (*.nl1 and *.doc.nl1)
Pattern Distribution	
After you have tested yo	our pattern plug-ins, package them into a pattern archive by clicking "Create Pattern Archive" and then distribute
To install the pattern, the	e pattern user must launch the pattern archive file in a web browser, launch the pattern archive file in a file explc
For more information abo	but packaging and distributing user-defined patterns, click here.

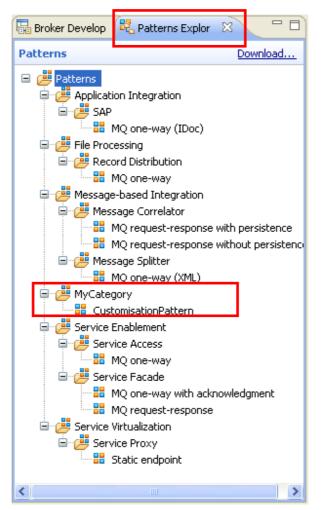
Source Files Pattern Configuration Categories Create Pattern

5. When the new Integration Toolkit instance starts accept the default workspace location and click OK.

🜔 Workspace Launcher
Select a workspace
WebSphere Message Broker Toolkit - Message Broker stores your projects in a forder called a workspace. Choose a workspace folder to use for this session.
Workspace: C:\Documents and Settings\wmbadmin\IBM\wmbt80\workspace Browse
Use this as the default and do not ask again OK Cancel

8. Instantiate the Modified Pattern

1. Go to the Pattern Explorer view, by clicking on its tab:

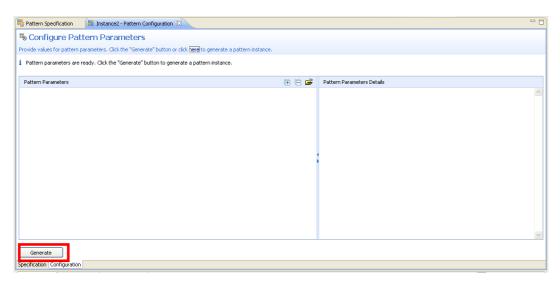


Double-click on CustomisationPattern to create a new instance. Enter "CustomisationTest2" as its name and click OK.

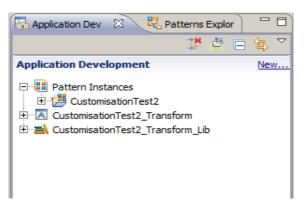
New Pattern Instance	<u>_ 0 ×</u>
Create New Instance The instance name uniquely identifies the pattern instance project that is created when the pattern instance is saved or generated. The name ideally reflects the business and integration functions that the pattern supports.	
Pattern instance name: CustomisationTest2	
Э ОК	Cancel

Pattern Authoring – User Customisation Provided by IBM BetaWorks

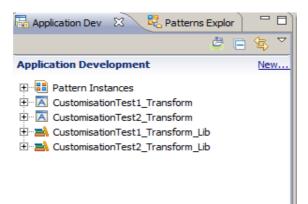
2. Click on the Generate button.



3. The new pattern instances will be created with the prefix CustomisationTest2. Note that the navigator has filters applied which will show only the newly generated projects.



You can remove the filters by clicking the icon at the top of the navigator (red cross). If you do this, you will see the generated application and library from CustomisationTest1.



9. Customise the Message Flow again

1. In the CustomisationTest2 library, open the TransformSub1 subflow, delete the Passthrough node and replace it with a Compute node, as before.

Connect the nodes as the follows.

🖥 Application Dev 🕺 🤻 Patterns Explor 🛛 🗖 🗖	🔡 CustomisationTest1 - P 🛛 📲 Pattern Specification 🖉 CustomisationTest2 - P 🖓 TransformSub1.subflow 🕅	-
Control C	Image: Services Image: Services Sca Sca WebSphere Adapters Compute Image: Routing Image: Sca WebSphere Adapters Compute Imansformation Construction Construction Construction Properties Properties Image: Sca Problems Image: Sca Properties Image: Sca Propertis Image: Sca	<u>a</u> 7 E

Double-click on the Compute node to generate the default ESQL file.

Save the message flow by pressing Ctrl+S or File->Save and close it.

10. Regenerate the Pattern Instance again

1. Now you will re-generate the pattern instance and check that the customisation you've made to the subflow isn't overwritten.

Go back to "CustomisationTest2 - Pattern Configuration".

Battern Specification	🔠 CustomisationTest2 - Pattern Configuration 🛛	🛛 💷 TransformSub1.subflow			
Sconfigure Pattern Parameters					
Provide values for pattern parameters. Click the "Generate" button or click here to generate a pattern instance.					
i Pattern workspace resources are generated successfully!					
Pattern Parameters		🕀 🕞 🗃 Pat			
		•			
Generate					
Specification Configuration					

2. Click on the Generate button.

A popup window will appear asking if you want to overwrite the existing instance. Click OK.

Pattern Instance Generation		
1	The project Instance2_TransformLib, Instance2_TransformApp, Instance2_Transform cannot be generated because it exists in your workspace. Click OK to delete the project.	
	OK Cancel	כ

3. Now open the generated TransformSub1 subflow again.

==	CustomisationTest2 - Pattern Configuration	💷 TransformSub1.subflow 🛛
	Input Compute	PO

Note that after re-generating the pattern instance, the customised subflow kept all the changes you've made, (demonstrated by the fact that the Compute node is still there).

This concludes the Customisation section of the Pattern Authoring lab.