Content Based Routing Message Flow

Overview:

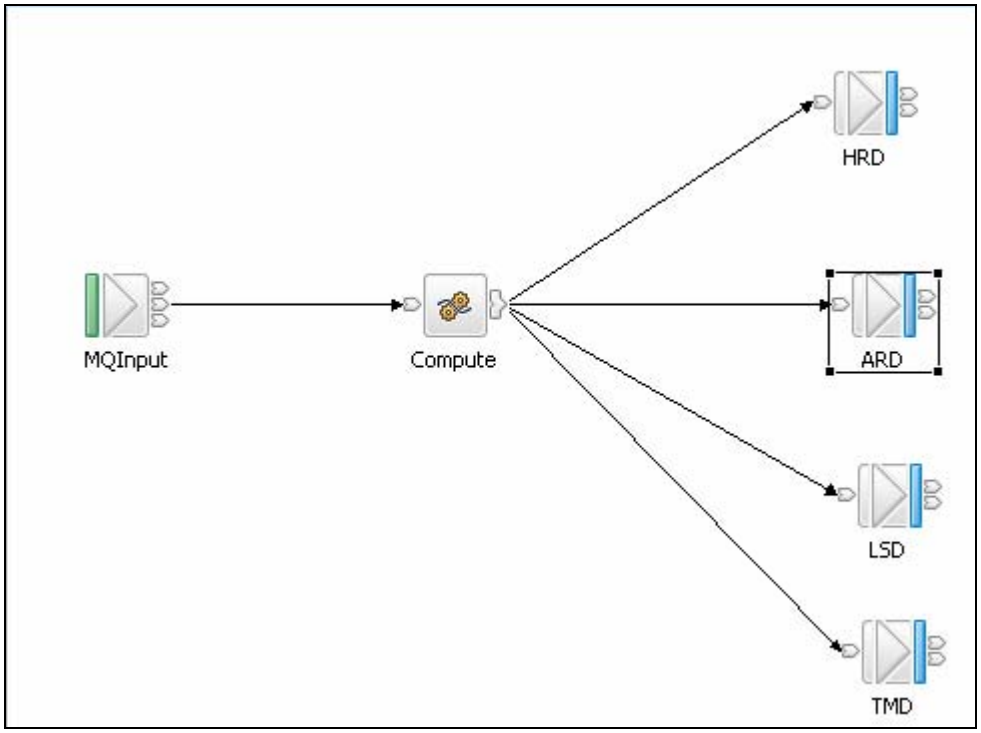
We had a client and that client required to meet the following scenario. We used Web Sphere Message BrokerV6.1 to achieve the solution. This document will explain how we achieved the desired solution.

Scenario:

Scenario was that there was a website that used to generate a message in its Central Application that was a Web Sphere MQ based and that message required to be transmitted to its destination.

There were four candidate departments to receive the message mainly HRD, ARD, LSD and TMD.

The scenario can best be understood with the following figure.



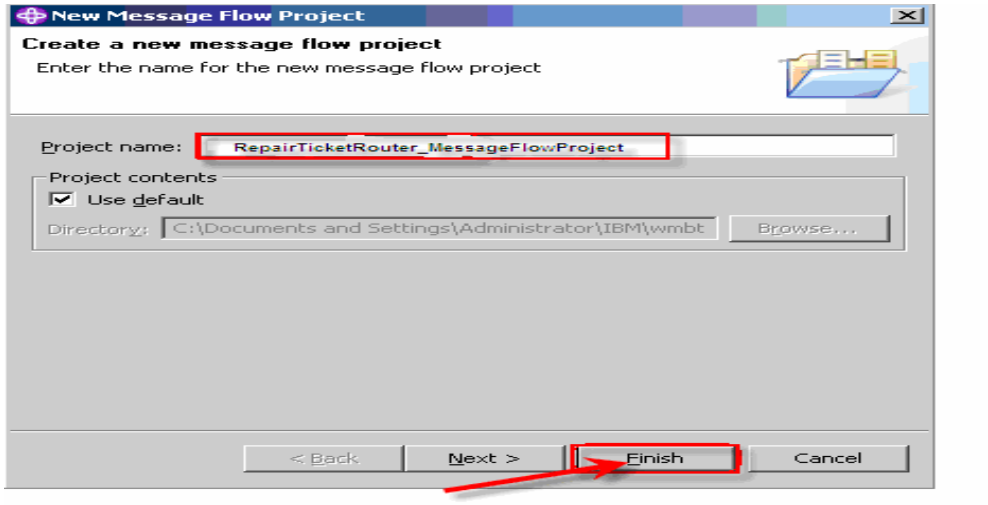
Above figure suggest that a message come from Input source that is any input queue and that goes in the central application and which department will receive that message depends on the contents of that message.

Solution:

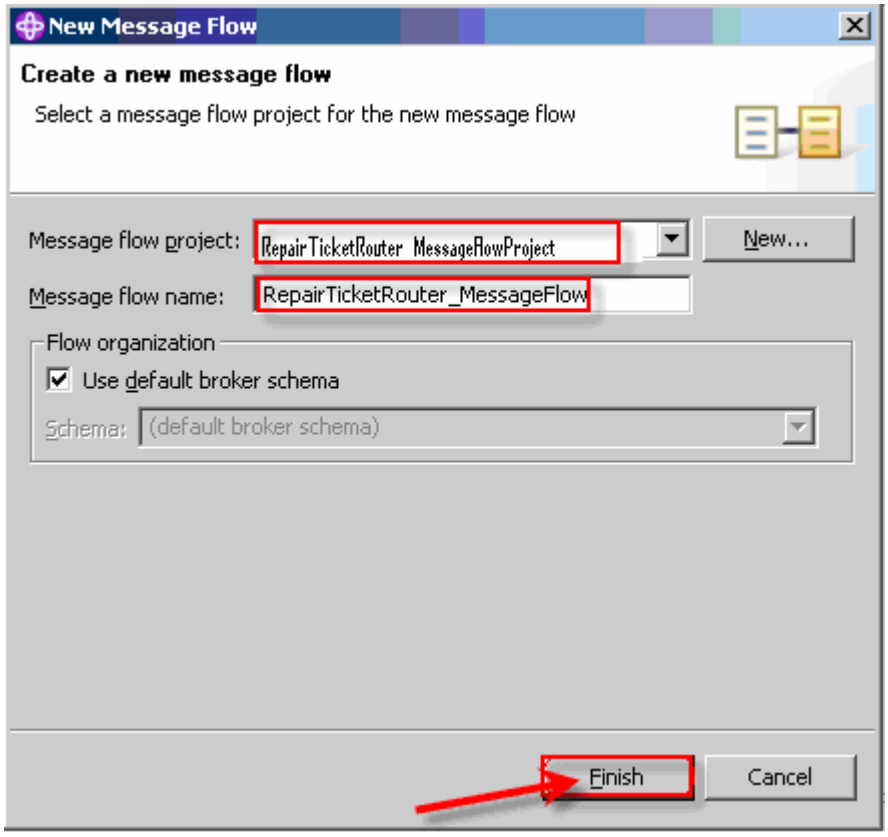
Websphere Message Broker facilitate us that if we put message in a XML format or in any other format in the input node and after that message goes to the central application that is broker and on the contents of that message our central application will route the message to specific department.

We will discuss in this paper how the above solution is achieved using Web Sphere Message Broker.

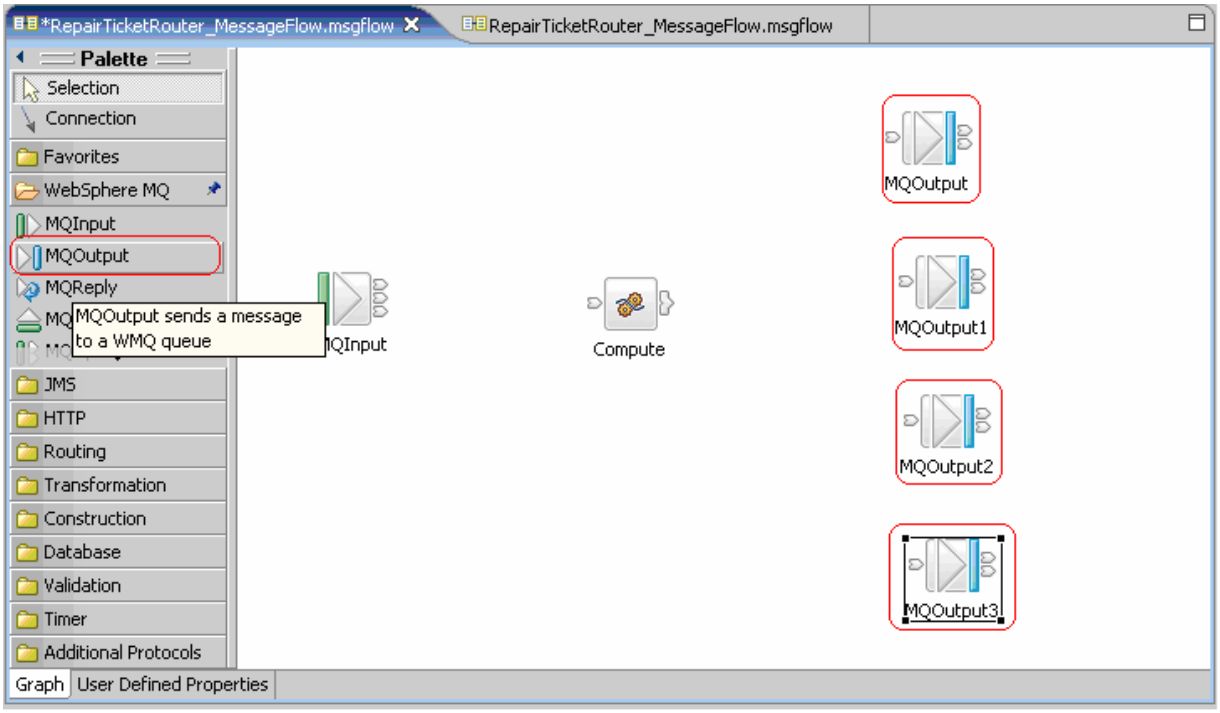
► First create a Message Flow Project.



► After creating the Message Flow Project create the Message Flow inside the Message Flow Project.



► Take the MQInput , MQOutput and Compute nodes from the Palette and drop on the canvas as shown below.

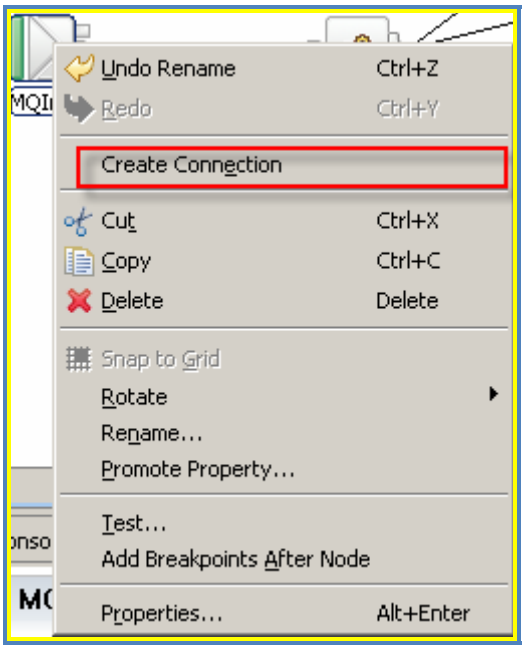


► Next rename each node as below:

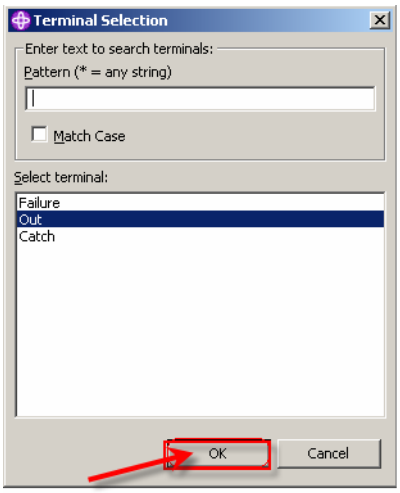
Node Type	Node Name
Compute	Compute
MQInput	MQInput
MQOutput	HRD
MQOutput1	ARD
MQOutput2	LSD
MQOutput3	TMD

► Define the order in which the nodes process an input message by connecting them together as:

► Right-click the **MQInput** node, then click **Create Connection**.



► In the Select Terminal dialog, click **Out**, then click **OK**. An arrow from the **MQInput** node follows the mouse pointer when you move the mouse because you have not specified which node to connect to.



► Click the **Compute** node. The arrow connects the **MQInput** node to the **Compute** node.

► Right-click the **Compute** node, then click **Create Connection.**

► In the Select Terminal dialog, click **Out**, then click **OK**. An arrow from the **Compute** node follows the mouse pointer when you move the mouse because you have not specified which node to connect to click **HRD** node. The arrow connects the Compute node to the **HRD** node.

► Right-click the **Compute** node, then click **Create Connection.**

► In the Select Terminal dialog, click **Out1**, and then click **OK**. An arrow from the **Compute** node follows the mouse pointer when you move the mouse because you have not specified which node to connect. Click **ARD** node. The arrow connects the Compute node to the **Plumbing** node.

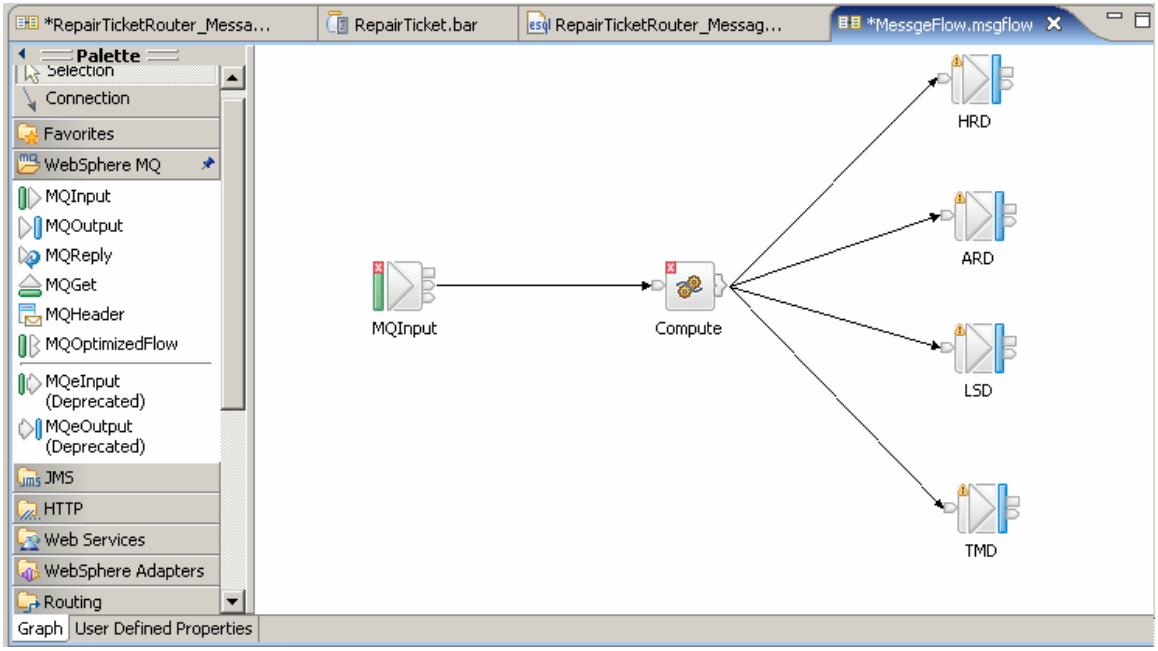
► Right-click the **Compute** node, then click **Create Connection.**

► In the Select Terminal dialog, click **Out2** and then click **OK**. An arrow from the **Compute** node follows the mouse pointer when you move the mouse because you have not specified which node to connect Click **LSD** node. The arrow connects the Compute node to the **LSD** node.

► Right-click the **Compute** node, then click **Create Connection.**

► In the Select Terminal dialog, click **Out3**, and then click **OK**. An arrow from the **Compute** node follows the mouse pointer when you move the mouse because you have not specified which node to connect Click **TMD** node. The arrow connects the Compute node to the **TMD** node.

When you save a message flow file, the Message Flow editor validates the message flow. **MessageFlow** has two errors, as shown in the following figure which are indicated by a small white cross on a red background on the MQInput and Compute nodes.



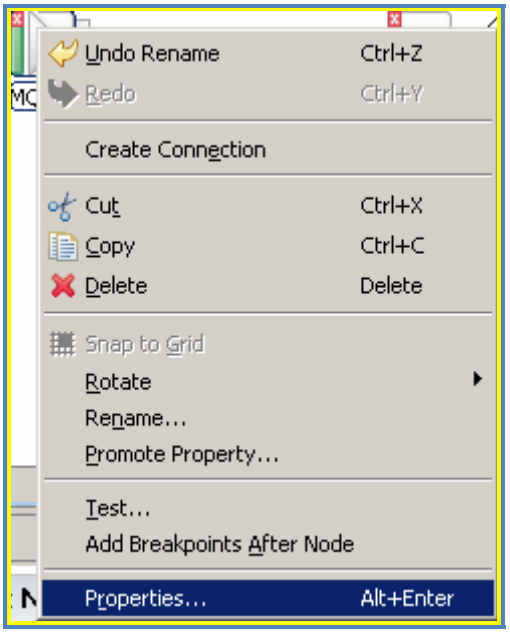
A brief description of each error is given below:
►The error in the MQInput node is because you have not entered the name of the Web Sphere MQ input queue from which the MQInput node takes input messages.

► The error in the Compute node is because you have not created the ESQL module that defines how the Compute node should process input messages.

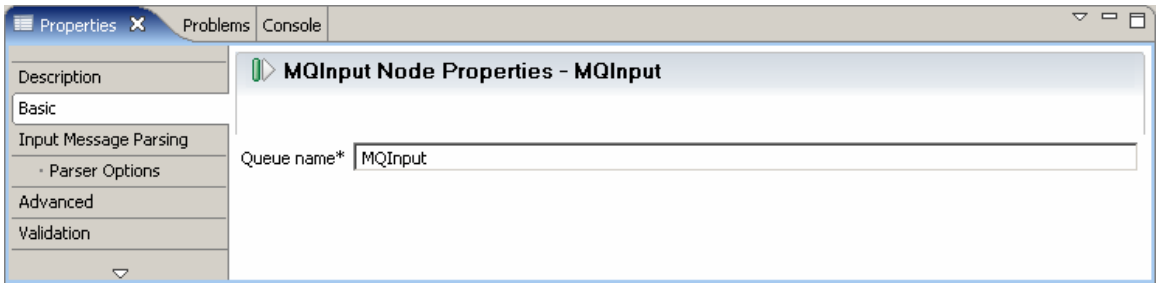
► To set the properties of the nodes in the MessageFlow:

► Make sure that the **Selection** button at the top of the node **palette** is highlighted so that you can select nodes on the canvas.

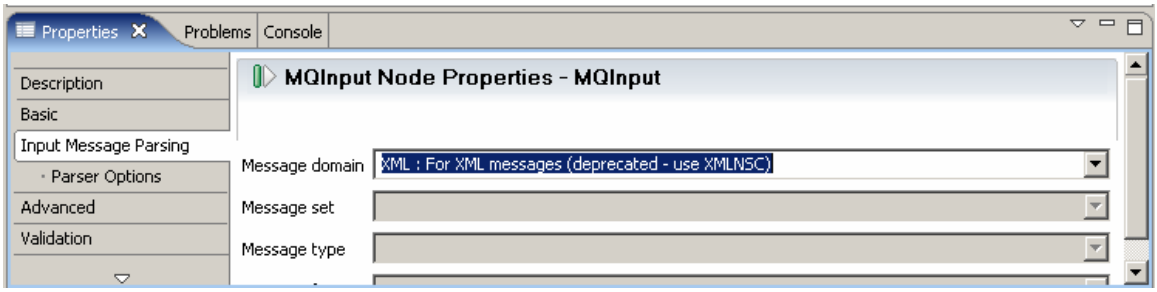
b. Right click on the **MQInput** node, then Click **Properties**



► In the properties page select Basic and insert the name of input queue in our scenario input queue name is MQInput.



► In the **properties** page select **Input Message Parsing** and then insert value **XML** in the **Message domain** field.



► Right click on the **HRD** node, then Click **Properties**

► In the **properties** page select **Basic** and insert **output queue name that will get message for HRD department in our scenario the name of output queue was HRD**.

► Right click on the **ARD** node, then Click **Properties**

► In the **properties** page select **Basic** and insert **output queue name that will get message for ARD department in our scenario the name of output queue was ARD**.

► Right click on the **LSD** node, then Click **Properties**

► In the properties page select Basic and insert output queue name that will get message for LRD department in our scenario the name of output queue was LSD.

h. Right click on the **TMD** node, then Click **Properties**

► In the properties page select Basic and insert output queue name that will get message for TMD department in our scenario the name of output queue was TMD.

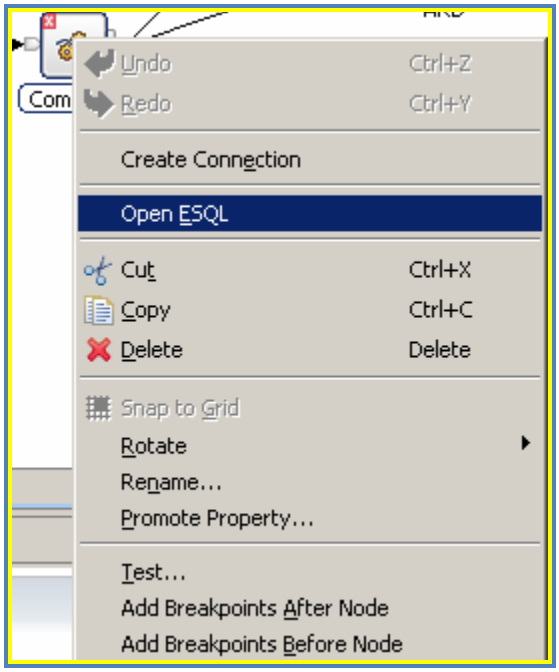
► **Save the MessageFlow.msgflow**

Note that in the Message Flow editor, the error indicator on the MQInput node is no longer displayed.

► **Writing ESQL for the Compute node**

All of the **ESQL** that belongs to a message flow is stored, by default, in a single file. In this case, all of the ESQL for the **MessageFlow** is stored in a file called **MessageFlow.esql**.

a. In the Message Flow editor, right-click the **Compute** node, then click **Open ESQL.**



Here is the description of ESQL code segments that we have coded inside the **MessageFlow.esql**.

We have created following two variables inside the main function.

DECLARE **Dept** CHARACTER;
DECLARE **I** INTEGER;

We have also created the **IF ELSE** structure that helps to propagate message to the different terminal depending on the value of Dept variable. As those terminals point to different output nodes and those nodes are configured with different department queues.

```
IF Dept = 'A' THEN
PROPAGATE TO TERMINAL 'out1';
END IF;

IF Dept = 'L' THEN
PROPAGATE TO TERMINAL 'out2';
END IF;
IF Dept = 'T' THEN
PROPAGATE TO TERMINAL 'out3';
END IF;
IF Dept = 'H' THEN
```

```
            PROPAGATE TO TERMINAL 'out';
            END IF;
            RETURN FALSE;
    END;
```

Now you can create the bar file that will contain our designed Message Flow and then deploy the bar file on execution group of broker.

IBM Global Services
Route 100
Somers, NY 10589
U.S.A.
Produced in the United States of America
08-10