Data Sync Implementation Notes

1. Setup Changes

* Table modifications

CREATE TABLE LEGACY\_MAPPING

("LEGACY\_MAPPING\_ID" NUMBER(19,0) PRIMARY KEY,

"SYSTEM\_NM" VARCHAR2(200 CHAR),

"CODE\_TYPE" VARCHAR2(200 CHAR),

"INTERNAL\_CODE" VARCHAR2(200 CHAR),

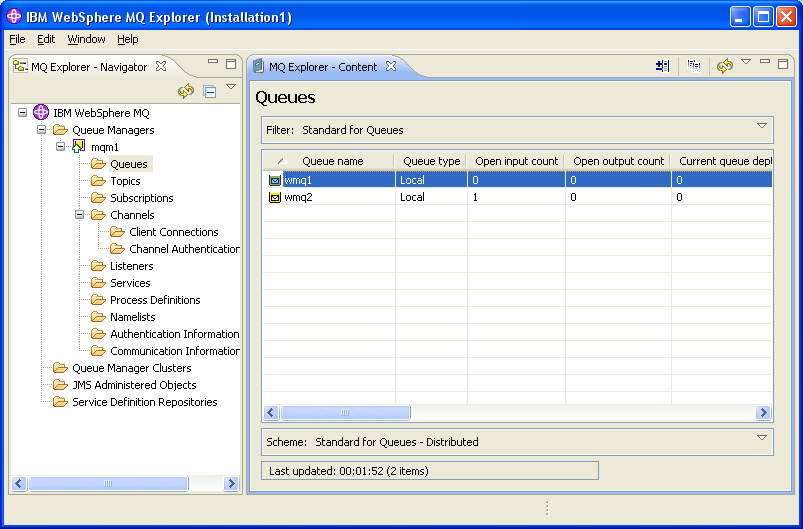
"EXTERNAL\_CODE" VARCHAR2(200 CHAR)

);

alter table ENTITY add LEGACY\_ID VARCHAR(9);

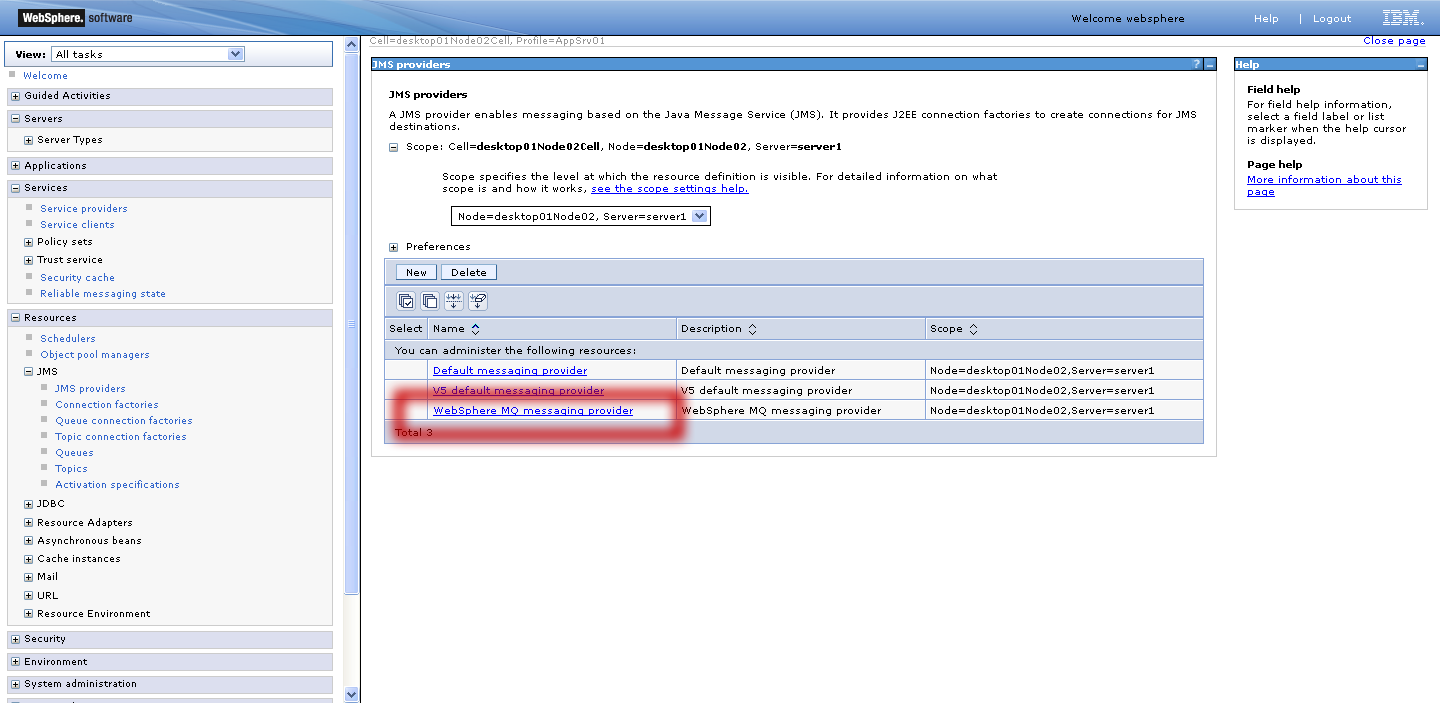
* MQ Setup

You need 2 queues, 1 for transmission, 1 for receiving

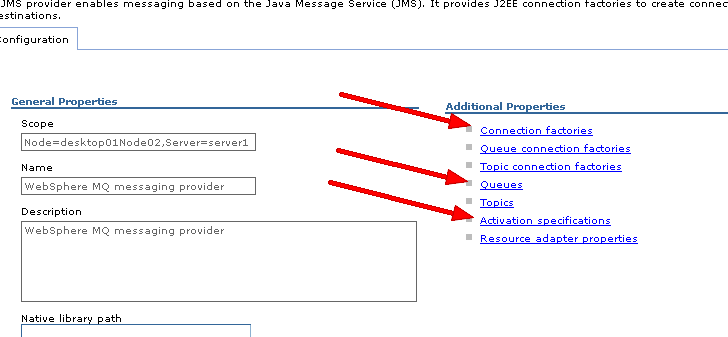


* Setup WAS -> MQ connectivity

Open Resources > JMS then select the MQ provider



Create a Connection Factory and 2 Queues (to connect to your MQ) and the activation spec (to trigger the message driven bean)



Update /cms-portal/cms-business-process/src/main/resources/META-INF/ibm-ejb-jar-bnd.xml

With the JNDI names of your queue/activation spec

For example, if your activation is jms/wmqa and your receive queue is jms/wmq2:

<message-driven name="DataSynchronizationMessageBean">

<jca-adapter activation-spec-binding-name="jms/wmqa" destination-binding-name="jms/wmq2" />

</message-driven>

Now if your connection factory is jms/wmq and your transmit queue is jms/wmq1

<message-destination-ref

name="gov.medicaid.services.impl.ProviderEnrollmentServiceBean/dataSyncQueue"

binding-name="jms/wmq1" />

<resource-ref

name="gov.medicaid.services.impl.ProviderEnrollmentServiceBean/mqConnectionFactory"

binding-name="jms/wmq" />

1. Code Changes
   * For transmitting changes, the relevant update is in

ProviderEnrollmentServiceBean#sendSyncronizationRequest(long ticketId)

* + The export implementation is in

FlatFileExporter.java

* + For receiving, the message driven bean is

DataSynchronizationMessageBean

It accepts either a TextMessage or a BytesMessage and updates the Legacy ID

1. Logic
   1. Upon approval, the JBPM handler will call sendSyncronizationRequest(long ticketId).
   2. sendSyncronizationRequest will retrieve the approved ticket and create a FlatFileExporter
   3. The exporter will generate a single row based on the provided record-mapping.xlsx, it will also include the header. For lookup values, a code-mapping is used (the new table), If no external mapping is found, the internal code for the lookup is used.



| Add-Update ID| Oracle Row ID| Legacy ID | Sys Err Code | Data Err Code | <provider file>

Add-Update ID: 1 byte. A for add, U for update (existing)

Oracle Row ID: 22 bytes – our row ID for this provider

Legacy ID: 8 bytes – the legacy ID of the provider

Sys err Code: 2 bytes – we record this to the log for every processed row, but ignore it

Data Err Code: 30 bytes (which will be 3 byte segments)

Please note “|” separater is sent as part of the header, it is just a continuous string

* 1. DataSynchronizationMessageBean will receive the response, parse the header and update the Legacy ID