



Leading the **Integration** Revolution

Your business problems have changed.
Why hasn't your integration solution?



ADEPTIA

Using Adeptia

Use Case: Real-time Message Queue Trigger

Use Case highlights

Use Message Queue trigger to kick-off a process flow:

1. Real time trigger to execute an orchestration in Adeptia
2. Take the message from a queue, map and load it into a database
3. Track MQ Event firings and check the status of each transaction
4. View run-time transaction logs and dashboards for details

Design methodology

Design

Design a high level process flow

Configure

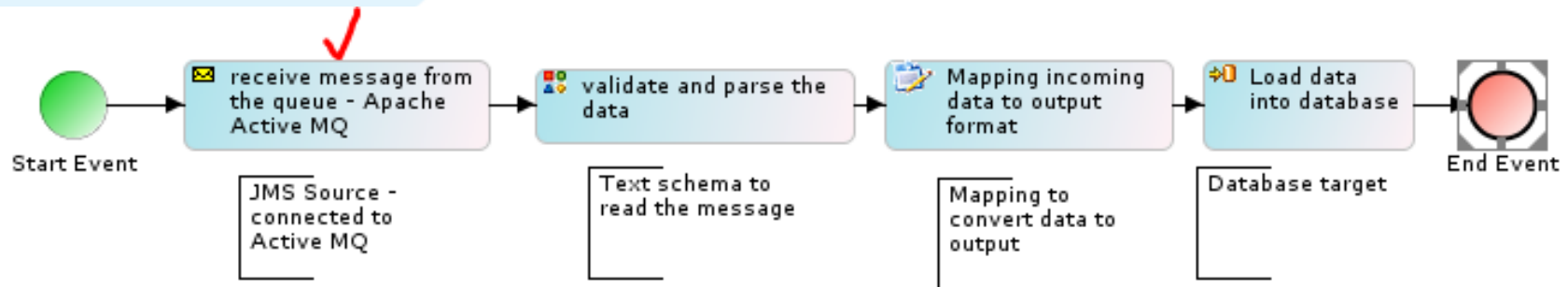
Configure all the activities in the process flow

Deploy

Attach to JMS Event

Design: Process Flow

Please refer to the Use Case video describing how to build this process flow



Configure – JMS Provider

First create a connection to MQ by going to Develop > Services > Connector > JMS Provider

Services > Connector > JMS Provider ✓

Delete + Create New Quick Search

Name	Description	Owner	Project Name	Modified	Action
ApacheMQ_connector ✓	connector to Apache MQ	demou...	UseCase_Mes...	12/27/...	☰

Standard Properties

Name * ApacheMQ_connector X

Description * connector to Apache MQ

JMS Server Type Apache ActiveMQ ✓

Provider Jar Files * activemq-all-5.4.2.jar Upload Jars

JNDI Factory * org.apache.activemq.jndi.ActiveMQInitialContextFactory

URL tcp://localhost:61616

Username

Password

Confirm Password

Queue Connection Factory * QueueConnectionFactory

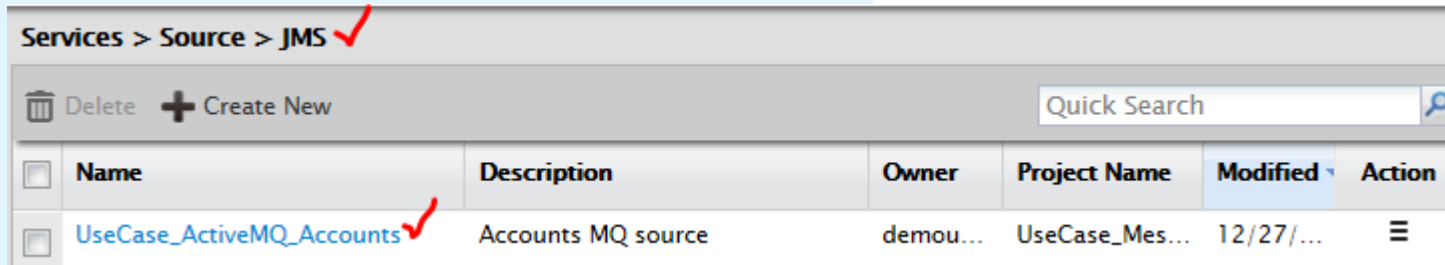
Topic Connection Factory * TopicConnectionFactory

Save Save As

Adeptia can connect to any MQ. Select the MQ from the JMS Server Type.

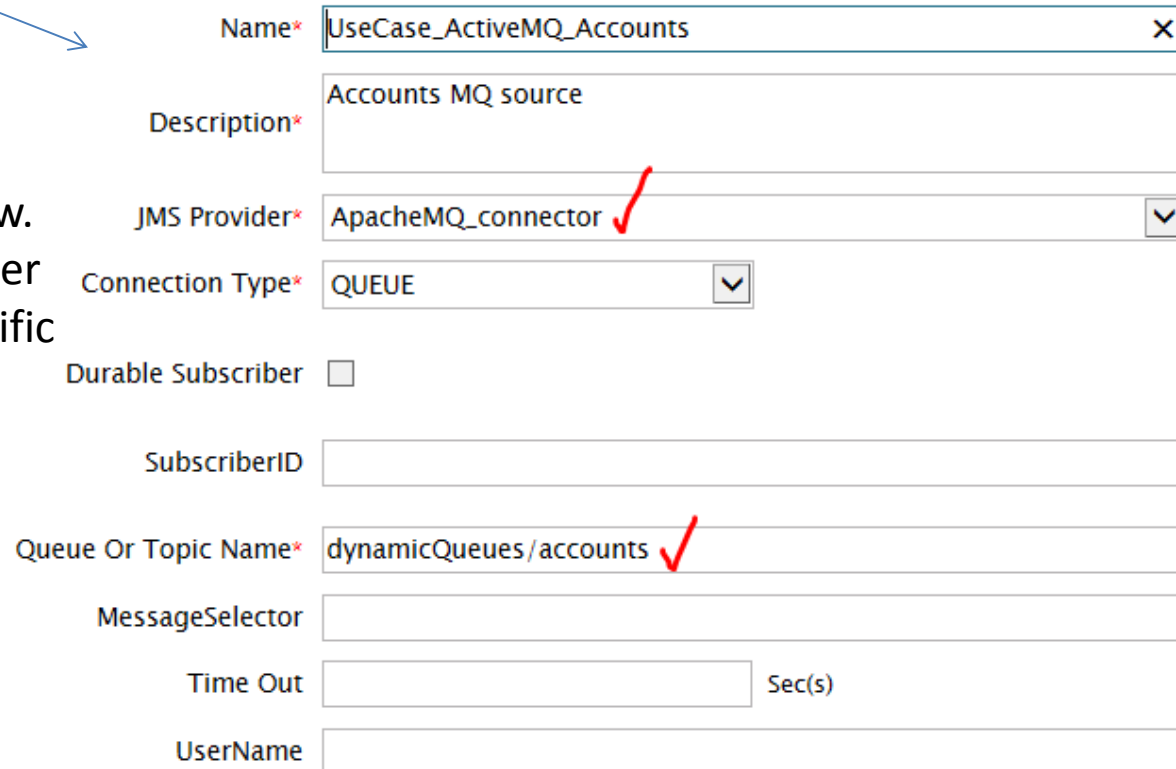
Configure – JMS Source

Create a JMS Source by going to Develop > Services > Source > JMS Source



Name	Description	Owner	Project Name	Modified	Action
UseCase_ActiveMQ_Accounts ✓	Accounts MQ source	demou...	UseCase_Mes...	12/27/...	☰

JMS Source is the first Activity in the process flow. Here we select the Provider and also provide the specific Queue name in MQ.



Name* UseCase_ActiveMQ_Accounts

Description* Accounts MQ source

JMS Provider* ApacheMQ_connector

Connection Type* QUEUE

Durable Subscriber

SubscriberID

Queue Or Topic Name* dynamicQueues/accounts

MessageSelector

Time Out Sec(s)

UserName

Configure – Message format

Create schema of the message format at Develop > Services > Schema > Text

Services > Schema > Text ✓

Delete + Create New Quick Search

Name	Description	Owner	Project Name	Modified	Action
UseCase_Accounts_queue_format	tab delimited data	demou...	UseCase_Mes...	12/27/...	☰

Edit Text Schema: UseCase_Accounts_queue_format

Data Header Present

Record Separator* \n

Field Separator* \t

Definition Mode* Import Definition File Enter the Fields Sequentially

Definition File Data Upload File

Similarly create the Database schema of the target. Refer to the Video.

#	Field Name	Type	Date Format	Time Format	Repeat
1	account	number	mmddyyyy	hh:mm:ss	1
2	name	string	mmddyyyy	hh:mm:ss	1
3	amount	number	mmddyyyy	hh:mm:ss	1
4	contact	string	mmddyyyy	hh:mm:ss	1
5	bank	string	mmddyyyy	hh:mm:ss	1

Number of Rows 1 at Position 6 Add Row Remove Row

Configure – Mapping

Create Mapping by going to Develop > Services > Data Transformation > Data Mapping

The screenshot displays the 'Services > Data Transformation > Data Mapping' configuration window. At the top, there is a breadcrumb trail and a 'Quick Search' field. Below this is a table listing mapping activities:

Name	Description	Owner	Project Name	Modified	Action
UseCase_Acouns_map	accounts map activity	demou...	UseCase_Me...	12/27...	⋮

The main workspace shows two XSD structures side-by-side:

- Structure_of_source_XSD:** Root (UseCase_Accounts_queue_format) containing a Record (M) with attributes: recordNumber, account (integer) (M), name (string) (M), amount (integer) (M), contact (string) (M), and bank (string) (M).
- Structure_of_destination_XSD:** Root (UseCase_Accounts_Queue_Target_schema) containing a Record (M) with attributes: Action, recordNumber, accountname (string) (M), sendingfacility (string), messagetype (string) (M), schemaname (string) (M), mapping (string) (M), expiration (string), and target (string) (M).

Blue lines indicate mappings from source to destination: recordNumber to recordNumber, account to accountname, name to messagetype, amount to schemaname, contact to mapping, and bank to target. A red line also connects the bank attribute to the target attribute.

At the bottom, the 'Mapping Rules' tab is active, showing a rule: `1 <S> bank..` mapped to `1 <T> target..`.

Configure – Database Target

Create Target by going to Develop > Services > Target > Database

Services > Target > Database ✓

Delete + Create New Quick Search

Name	Description	Owner	Project Name	Modified	Action
UseCase_Accounts ✓	accounts db target	demou...	UseCase_Mes...	12/27/...	☰

Edit Database Target: UseCase_Accounts

Standard Properties

Name* UseCase_Accounts ×

Description* accounts db target

Database Info* AccountsDB ▾

Schema Name* UseCase_Accounts_Queue_Target_schema ▾

Database Operation* Insert Update Insert/Update Update/Insert

Configure – Process Flow

The screenshot displays a process flow designer interface. On the left is the 'Repository View' with a tree structure of activities. A red arrow points to the 'JMS' folder. The main canvas shows a process flow starting with a 'Start Event', followed by a 'receive message from the queue - Apache Active MQ' activity (marked with a red checkmark), then 'validate and parse the data', 'Mapping incoming data to output format', 'Load data into database', and finally an 'End Event'. Below the canvas is a 'Properties' table for the selected activity.

Select the activities from the Repository View and attach to the steps in the flow. Also double click on the JMS Source and change its Event Context Enabled to True.

Name	Value
Activity Maximum Retries on Failure	0
Activity Wait Time(in seconds) between Retries	60
eventContextEnabled	true
ID	192168001039138816531999700024
Label	receive message from the queue - Apache Active MQ
Name	UseCase_ActiveMQ_Accounts
Process Flow Exit Control Message	
source	
streamNames	UseCase_ActiveMQ_Accounts
Synch	true
Type	JmsSource

Project: View and manage activities for this use case

Getting Started | Projects | Solutions ▾ | Process ▾ | Services ▾ | Events ▾ | Web Forms | Reports ar

Projects

List of activities included in project "UseCase_MessageQueue_trigger" ✓

◀ Back Move ↻ Quick Search

<input type="checkbox"/>	Name	Description	Owner	Modified
[-]	Data Mapping (1)			
<input type="checkbox"/>	UseCase_Acocunts_map	accounts map activity	demouser	12/27/13 11:38
[+]	Database Info (1)			
[-]	Database Schema (1)			
<input type="checkbox"/>	UseCase_Accounts_Que...	database schema	demouser	12/27/13 11:36
[-]	Database Target (1)			
<input type="checkbox"/>	UseCase_Accounts	accounts db target	demouser	12/27/13 11:38
[+]	Event Registry (1)			
[-]	JMS Event (1)			
<input type="checkbox"/>	Accounts_MQ_trigger	trigger to kick off proc...	demouser	12/27/13 11:44
[-]	JMS Provider (1)			
<input type="checkbox"/>	ApacheMQ_connector	connector to Apache MQ	demouser	12/27/13 11:26
[-]	JMS Source (1)			

Leading the **Integration** Revolution

Your business problems have changed.
Why hasn't your integration solution?

Thank You!



ADEPTIA