Deploying IBM MQ IVT wmq.jmsra.ivt.ear in WebSphere Application Server (WSAS) in Linux RHEL

https://www.ibm.com/support/pages/node/7148280

Date last updated: 14-Apr-2024

Rich Montjoy, Angel Rivera
IBM MQ Support
https://www.ibm.com/products/mq/support
Find all the support you need for IBM MQ

+++ Objective +++

The purpose of this tutorial is to configure and run the WebSphere Application Server (WSAS) in a Linux (RHEL) server, with the intention to run the IBM MQ Installation Verification Test (IVT) ear file, wmq.jmsra.ivt.ear.

For this tutorial:

- The version of MQ is 9.3.0.15 LTS, but the steps are applicable to all supported versions of MQ.

- The version of WSAS is 9.0.5.1 and has been already installed, but the steps are applicable to all supported versions of WSAS.

+++ About the MQ IVT application

1. The IVT sample is located in the MQ installation under the following directory (if the java client is installed):

<mq_install_dir>/java/lib/jca/wmq.jmsra.ivt.ear

2. It is a servlet that posts a message to an MQ queue and then uses an activation spec to consume the message.

3. This IVT sample is hard-coded to use a JMS Administrative object of type jmsQueue named IVTQueue.
This JMS administrative object will refer to the real (base) MQ queue named R93.LQ

4. The sample can use whatever server-connection channel that you wish.

++ References

https://www.ibm.com/docs/en/ibm-mq/9.3?topic=adapter-verifying-resource-installation
IBM MQ / 9.3
Verifying the resource adapter installation
The installation verification test (IVT) program for the IBM® MQ resource adapter is supplied as an EAR file. To use the program, you must deploy it and define some objects as JCA resources.

The following tutorial is old, but still applicable.
It shows the steps to create the Resources for the JNDI (Connection Factories, etc)

https://www.ibm.com/support/pages/node/322771
Using WebSphere MQ V7 as JMS Provider for WebSphere Application Server V7, V8.0 and V8.5, and MQ 9.0 for WAS V9.0

+++ MQ Queue Manager

++ Full MQ Server package downloaded from IBM Passport Advantage

+ To download the full MQ Server package, see:

https://www.ibm.com/support/pages/downloading-ibm-mq-93-lts
Downloading IBM MQ 9.3 LTS

For MQ 9.3: From IBM Passport Advantage you can download the manufacturing refresh:
MQ 9.3.0.15 LTS that includes 9.3.0.0 + 9.3.0.1 + 9.3.0.2 + ... + 9.3.0.10 + ... + 9.3.0.15

Visit IBM Passport Advantage:
https://www.ibm.com/software/passportadvantage/pao_customer.html
... search for Part Number for the single package that includes the queue manager, client and the rest of the components (except the MQ Explorer).

M0GCGML => IBM MQ 9.3.0.15 Long Term Support release for Linux on x86 64-bit Multilingual eImage

+ To install the MQ queue manager code, see:

The following tutorial can be helpful for the installation and uninstallation tasks when you want to have only 1 installation in the host. Keep in mind that the Fix Packs mentioned in the tutorial were the ones that were the latest at the time the tutorial was written.
Even though the titles specify MQ 9.3, the same principles apply to all versions, release types (LTS/CD) and fix packs.
It applies also to the queue manager and/or client components.
All the tasks are done with the default Installation1 which is /opt/mqm

https://www.ibm.com/support/pages/node/6988681
Installing MQ 9.3, applying Fix Pack 9.3.0.5, uninstalling in Linux RHEL

+ Location of the MQ IVT ear file

These are the locations for the MQ RA rar and IVT ear files.

Linux:
[mqm@WSAS-MQ1 mqm93]
$ cd /opt/mqm93/java/lib/jca/
$ ls -l
-r--r--r-- 1 mqm mqm   10262 Nov 29 02:06 wmq.jakarta.jmsra.ivt.ear
-r--r--r-- 1 mqm mqm   25604 Nov 29 02:06 wmq.jmsra.ivt.ear

For completeness:
Windows:
Directory of C:\Program Files\IBM\MQ\java\lib\jca
05/31/2023 01:00 AM   10,242 wmq.jakarta.jmsra.ivt.ear
05/31/2023 01:00 AM   25,617 wmq.jmsra.ivt.ear

+ These ear files are provided by rpm fileset in Linux:

rpm -qf /opt/mqm93/java/lib/jca/wmq.jmsra.ivt.ear
MQSeriesJava-9.3.0-15.x86_64

+ Local queue manager

Note: The default location for the MQ code is /opt/mqm, but in this tutorial a different location was used.
The RHEL host WSAS-MQ1 was used for this tutorial and MQ was installed in Installation1 at location:
/opt/mqm93

We will use a queue manager located in the same host of the WSAS server.

Queue manager name: R93
Host: localhost
Port: 1414
Server-Connection channel: R93.SVRCONN with MCAUSER('mqm') set.
Queue: R93.LQ

Because our focus is just to have a basic queue manager with minimum security for testing the MQ IVT application, the CONNAUTH and CHLAUTH are disabled.
One way to create such queue manager is by creating an mqsc file in /var/mqm such as:
/var/mqm/define-wsas-qm.mqsc

The contents of this mqsc file is:

+ begin of file (do not include this line)

```
** Define queues
DEFINE QLOCAL('R93.LQ')                              REPLACE

** Do not ask for password for remote access
ALTER AUTHINFO(SYSTEM.DEFAULT.AUTHINFO.IDPWOS) AUTHTYPE(IDPWOS) CHCKCLNT(OPTIONAL)
REFRESH SECURITY TYPE(CONNAUTH)

** Simplify the demo system by disabling channel and connection authentication
ALTER QMGR CHLAUTH(DISABLED) CONNAUTH(' ')
REFRESH SECURITY TYPE(CONNAUTH)

** Define channels to be used by a remote MQ client
```

```
DEFINE CHANNEL(R93.SVRCONN) CHLTYPE(SVRCONN) MCAUSER('mqm')
```

+ end of file (do not include this line)

Create the queue manager by issuing crtmqm with the flag -ic that points to the file that has the mqsc commands:

```
mqm@WSAS-MQ1.fyre.ibm.com:
$ crtmqm -u SYSTEM.DEAD.LETTER.QUEUE -p 1414 -ic /var/mqm/define-wsas-qm.mqsc   R93
$ strmqm R93
```

You need to ensure that the following command should work fine without errors, in that way, when you later on start and run the WSAS server, any new errors will be directly related to the use of WSAS and not from a basic initial configuration.

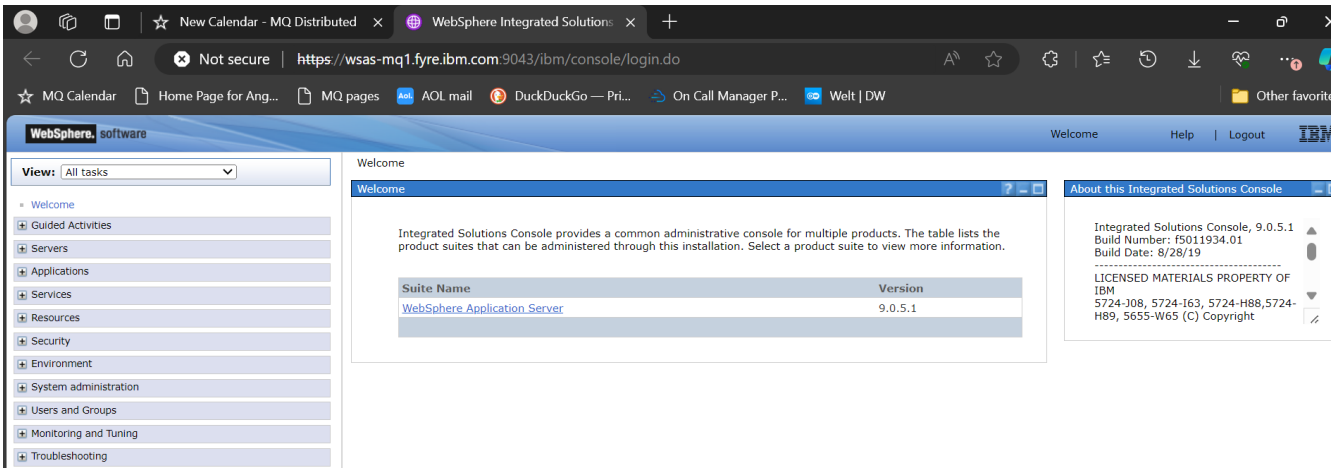
```
$ export MQSERVER='R93.SVRCONN/TCP/localhost(1414)'
$ amqsputc R93.LQ R93
Sample AMQSPUT0 start
target queue is R93.LQ
testing prior to start the WSAS server
Sample AMQSPUT0 end
```

+++ WebSphere Application Server

To start server1 for the profile AppSrv01 issue:
[root@WSAS-MQ1]
cd /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin
./startServer.sh server1

The log files will be located at:
/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/logs/server1

The WSAS Web Console can be accessed via:
https://wsas-mq1.fyre.ibm.com:9043/ibm/console/login.do



The MQ Resource Adapter (RA), that is, the MQ JMS code, is provided by WSAS.
The specific version of the MQ RA depends on the specific version of WSAS.
See the following article for more details.

https://www.ibm.com/support/pages/node/86587
Which version of WebSphere MQ Resource Adapter (RA) is shipped with WebSphere Application Server?

Because the version of WSAS is 9.0.5.1, then the included version of the MQ RA is:
WSAS 9.0.5.1 => MQ RA 9.1.0.2 (9.1.0.2-p910-002-190322)

WebSphere Application Server V9.0

WebSphere Application Server Version 9.0 ships with the WebSphere MQ Version 9.0 Resource Adapter. The table below shows the level of the Resource Adapter that is included with specific releases of the application server.

WebSphere Application Server Version	WebSphere MQ JCA resource adapter Version	Implementation Version, shown in WMSG1703I log entry during server startup
9.0.5.1	9.1.0.2	9.1.0.2-p910-002-190322

OK. At this point we have the 2 components running: the WSAS server and the MQ queue manager.

Now let's proceed to configure WSAS for the MQ IVT application.

+ Step 1: Create JMS Administrative objects

In WSAS you need to create 4 jndi resources at the cell scope.

The jndi names are hard coded in the MQ IVT application:

- A Connection Factory with a jndi name of: jms/ivt/IVTCF
- A Connection Factory with a jndi name of: jms/cf1
- An Activation Specification with a jndi name of: jms/ivt/IVTACT
- A queue with a jndi name of: jms/ivt/IVTQueue

+ Creating the Connection Factories

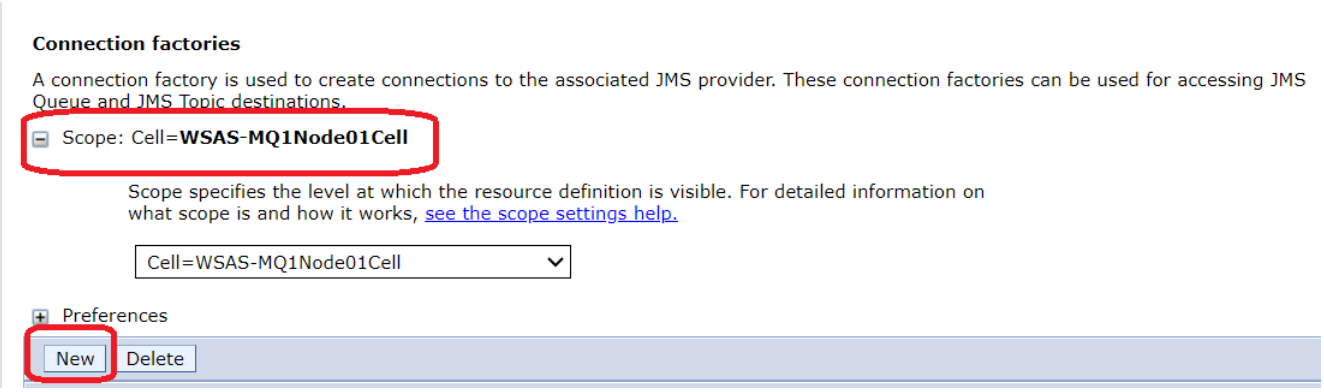
Start with the WSAS Web Console
https://wsas-mq1.fyre.ibm.com:9043/ibm/console/login.do

On the left navigational panel select:

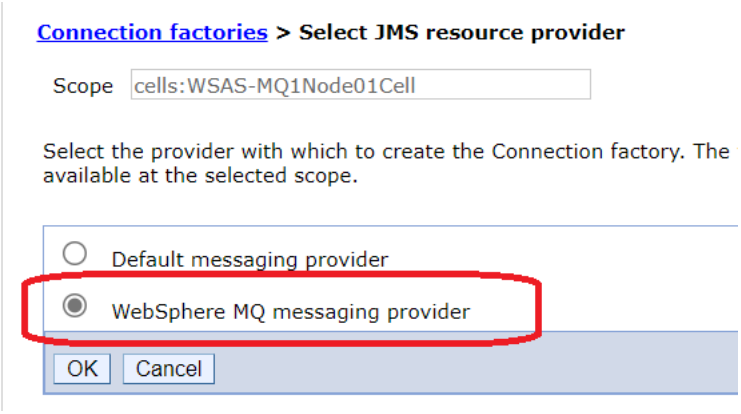
Resources > JMS

We will start with: Connection factories

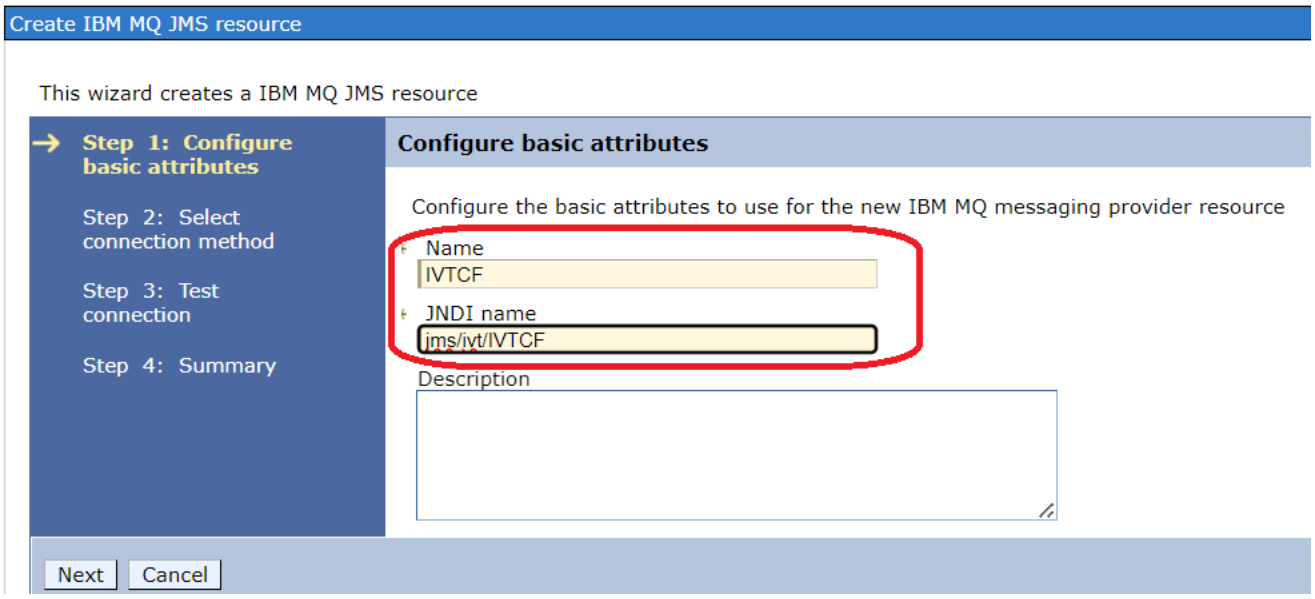
On the right navigational panel, accept the default scope of: Cell
Then click on: New



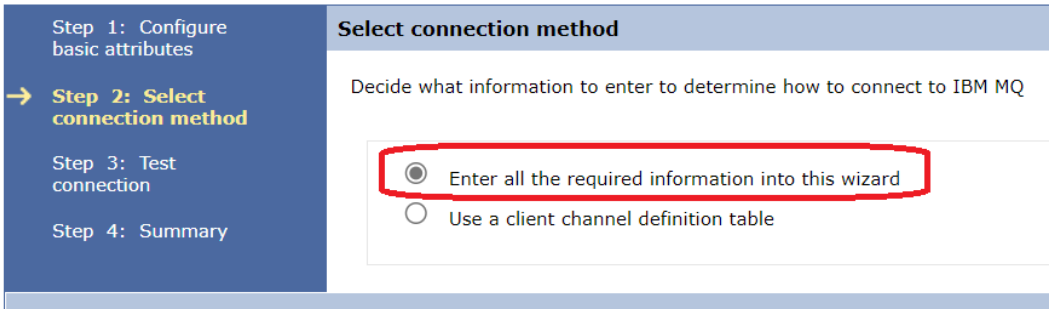
Under: Select JMS resource provider
… select:
(*) WebSphere MQ messaging provider



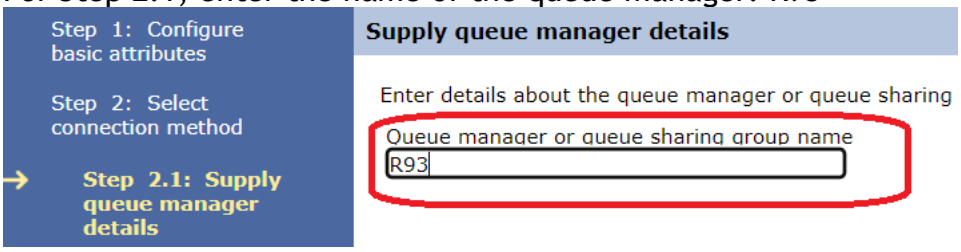
For Step 1, enter the basic attributes:
Name: IVTCF
jndi name of: jms/ivt/IVTCF



For Step 2, select:
(*) Enter all the required information into this wizard

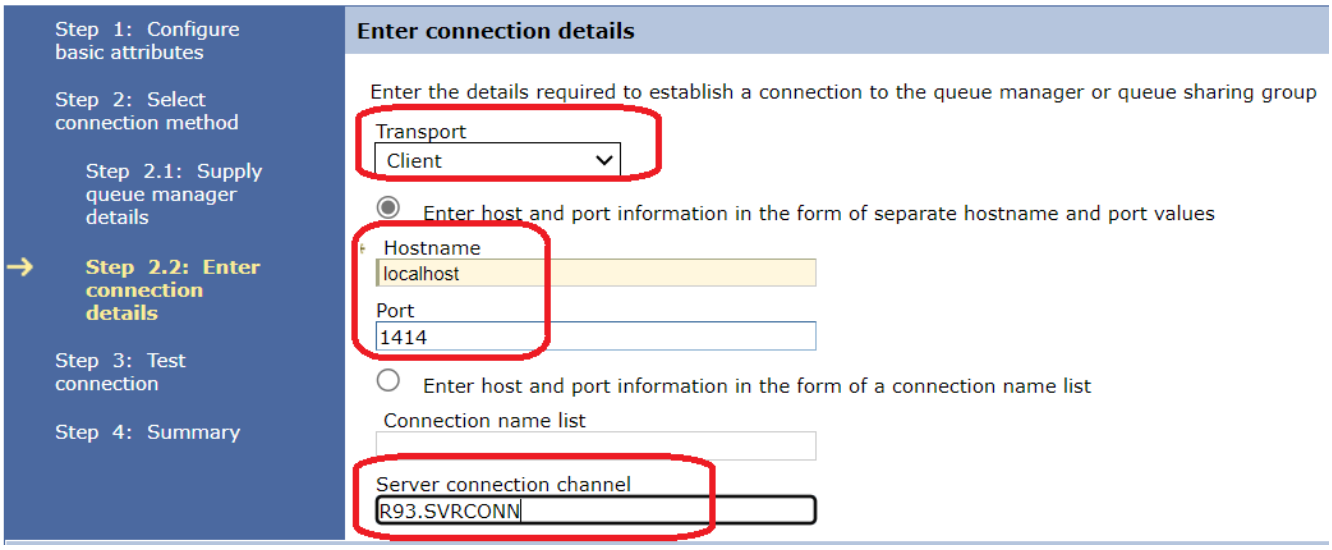


For Step 2.1, enter the name of the queue manager: R93

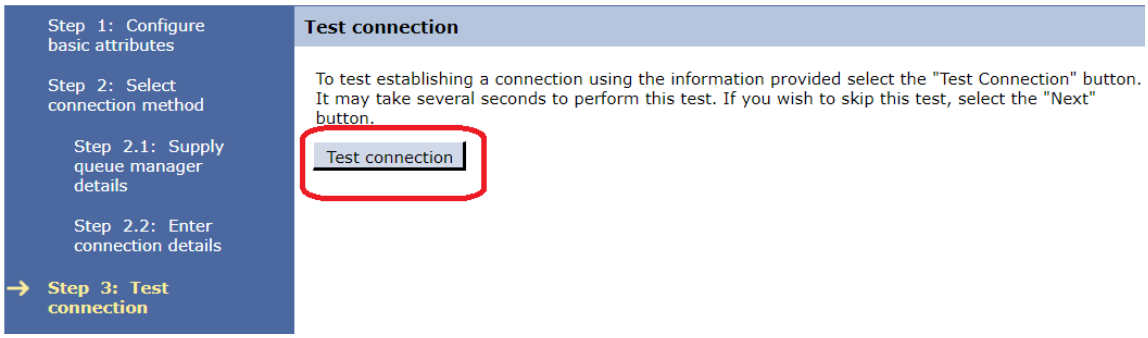


For Step 2.2, enter the details

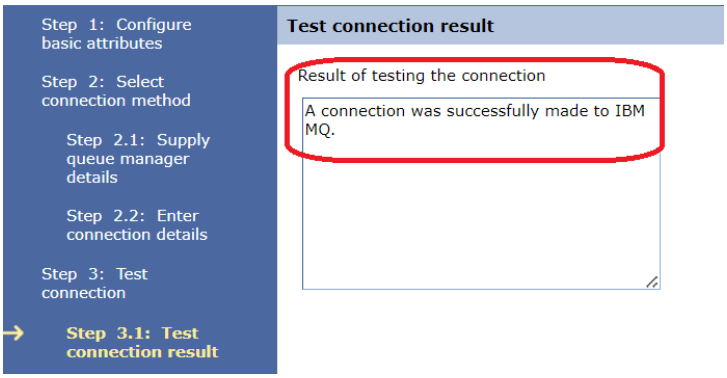
Transport: Client
Hostname: localhost
Port: 1414
Server connection channel: R93.SVRCONN



For Step 3, it is highly recommended that you actually test the connection!

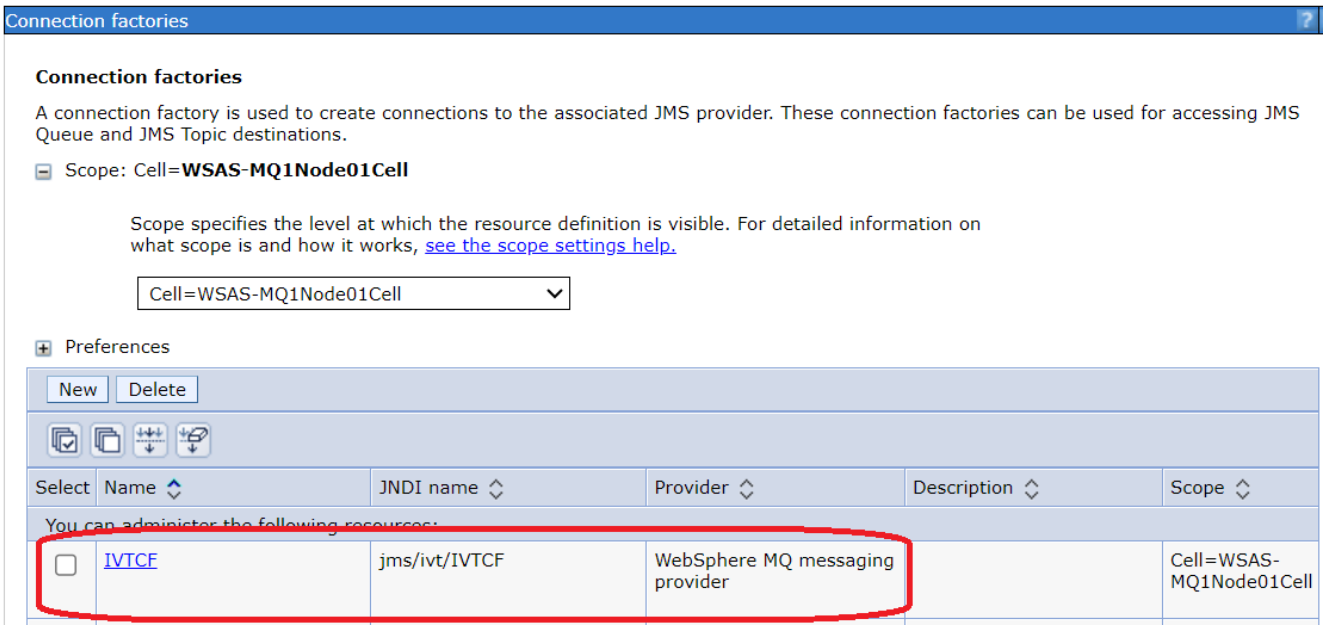


You can proceed only if the test was successful.



Continue with the steps.

You should see now the new entry for the Connection Factory: IVTCF

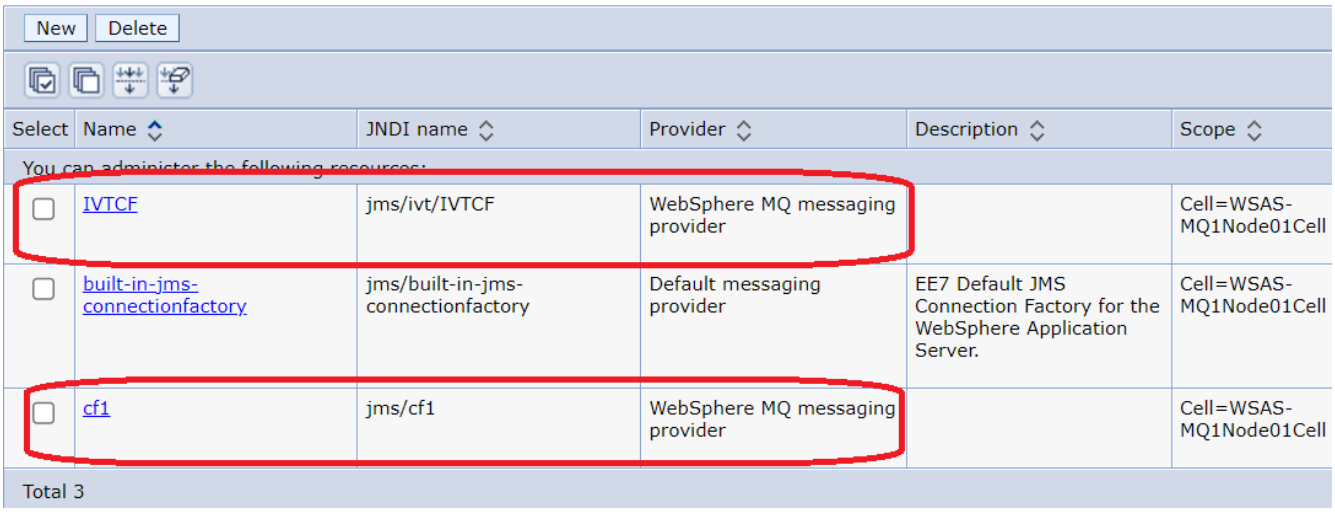


Repeat the steps to create the next Connection Factory: cf1

Name: cf1
jndi name of: jms/ivt/IVTCF

… with the same connection details.

Now we have a 2nd Connection Factory: cf1



Let's create a Destination Queue.

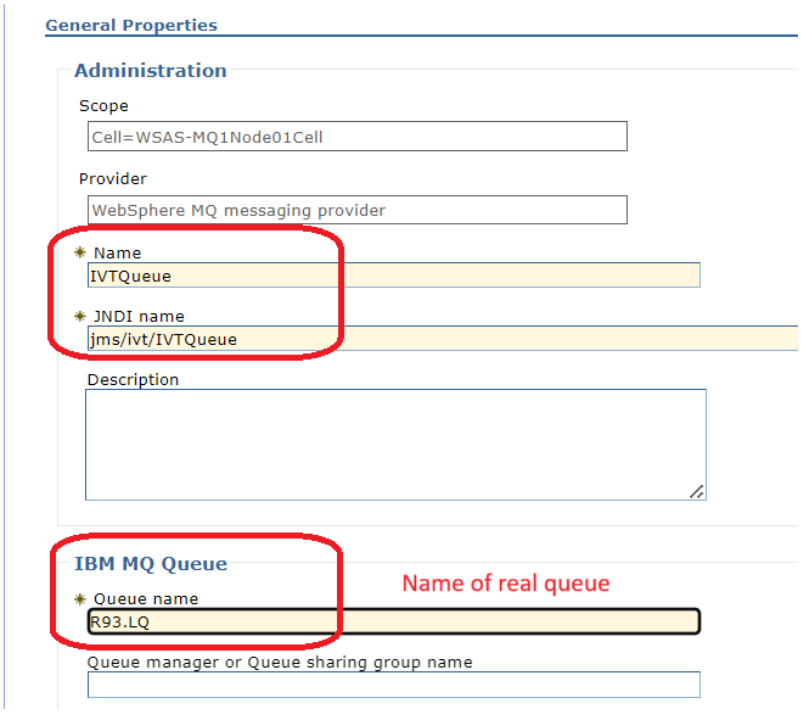
From the left navigation panel, select:

Resources > JMS > Queues

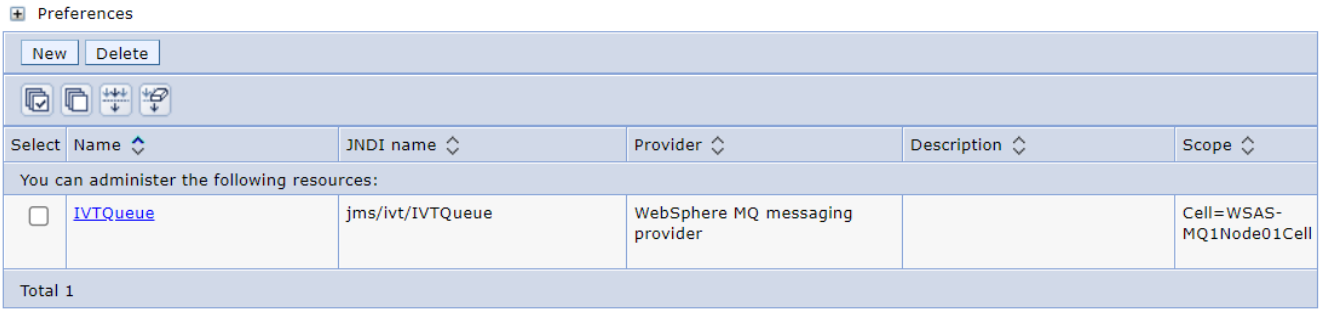
Proceed to specify that the provider is:
(*) WebSphere MQ messaging provider

… and the rest of the information:

Name: IVTQueue
Destination jndi: jms/ivt/IVTQueue
Name real queue:



You will see:



Let's create an Activation Specification.
From the left navigation panel, select:

Resources > JMS > Activation specificatons

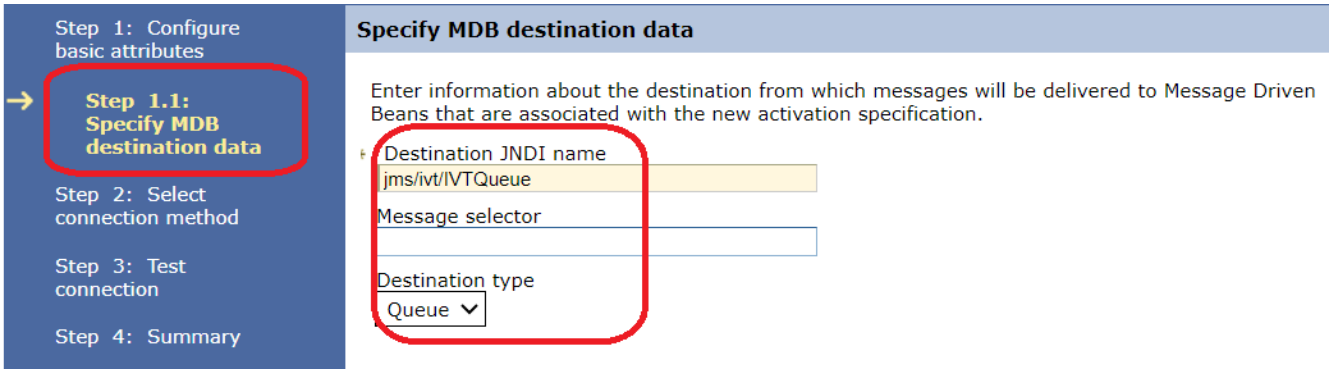
On the right navigation panel click: New

Proceed to create a new Activation Specification with the following values:

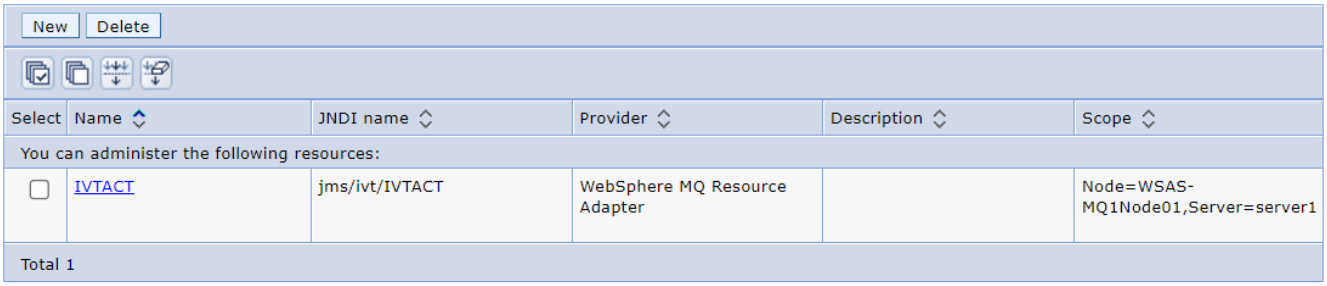
Name: IVTACT
jndi name of: jms/ivt/IVTACT
Destination jndi: jms/ivt/IVTQueue

The steps are very similar to the creation of a Connection Factory, with the exception that a Destination (queue) needs to be specified.

But instead of the name of the queue, it is necessary to specify the "jndi name":
jms/ivt/IVTQueue



Proceed with the prompts.
We will see a new entry.

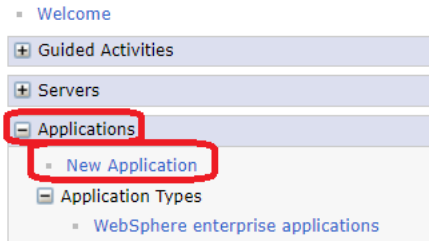


+ Step 2: Deploy the MQ IVT Enterprise Application

Now that we have defined the JNDI Resources, let's proceed to install and deploy the IVT application.

On the left navigation panel of the WSAS Console, select:

Applications > New Application



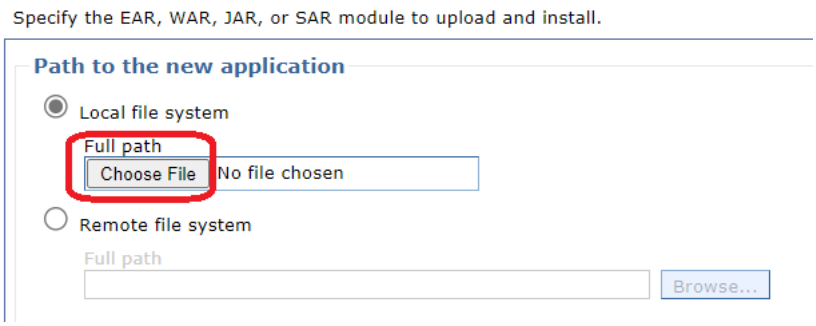
On the right navigation panel select:
New Enterprise Application

Specify the full path name of: wmq.jmsra.ivt.ear

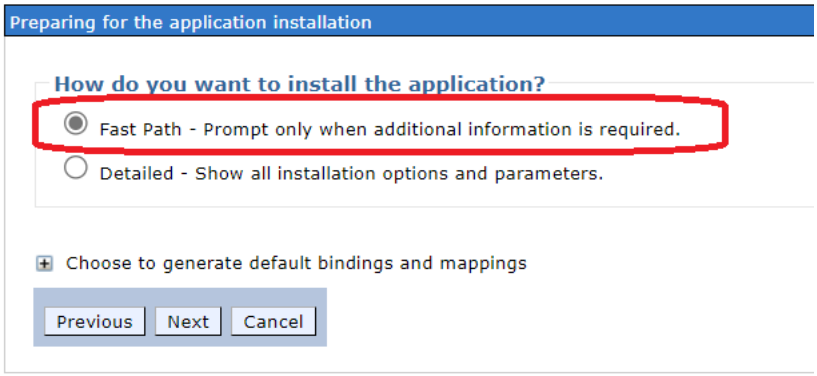
The term "Local file system" refers to the LOCAL HOST of the WEB BROWSER!
For example, if your browser is in Windows, the "local file system" would be the one provided by our MQ queue manager installation in Windows:
C:\Program Files\IBM\MQ\java\lib\jca\wmq.jmsra.ivt.ear

In Linux would be:
/opt/mqm/java/lib/jca/wmq.jmsra.ivt.ear

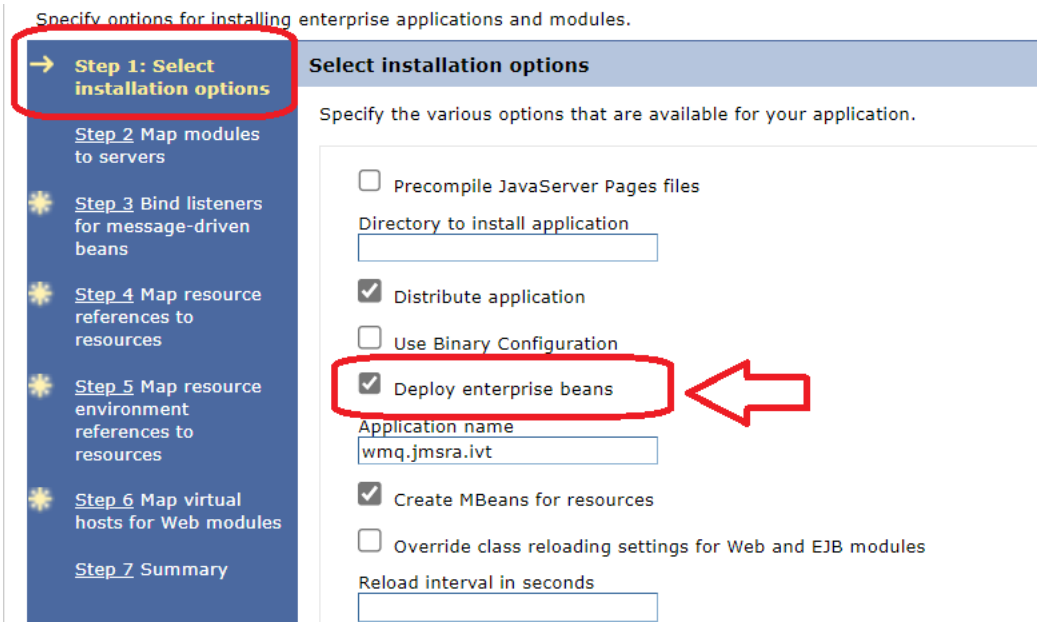
You can copy the IVT ear to a path under the WSAS directory (which will be the "remote file system" in the dialog.
For example:
/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/installedApps/WSAS-MQ1Node01Cell/wmq.jmsra.ivt.ear



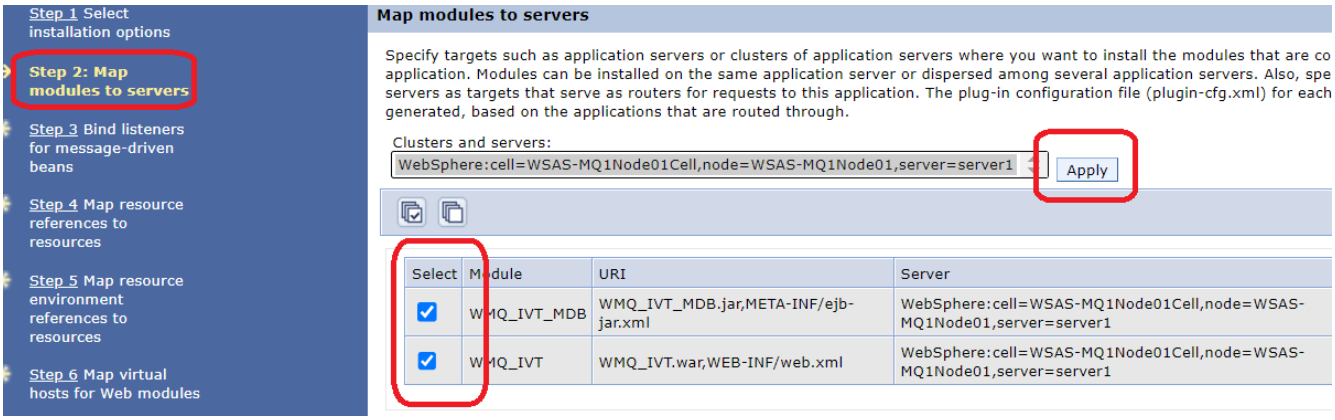
Accept the default:
(*) Fast Path - Prompt only when additional information is required



In "Step 1: Select installation options" you need to enable the checkbox:
(*) Deploy enterprise beans

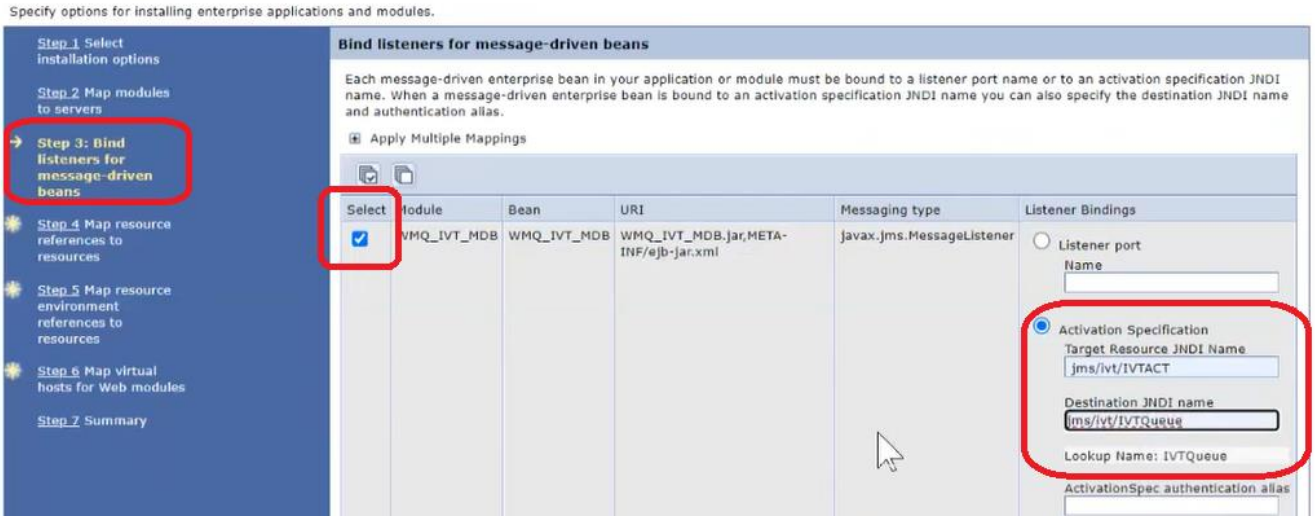


In "Step 2: Map modules to servers" enable the check boxes and click "Apply"



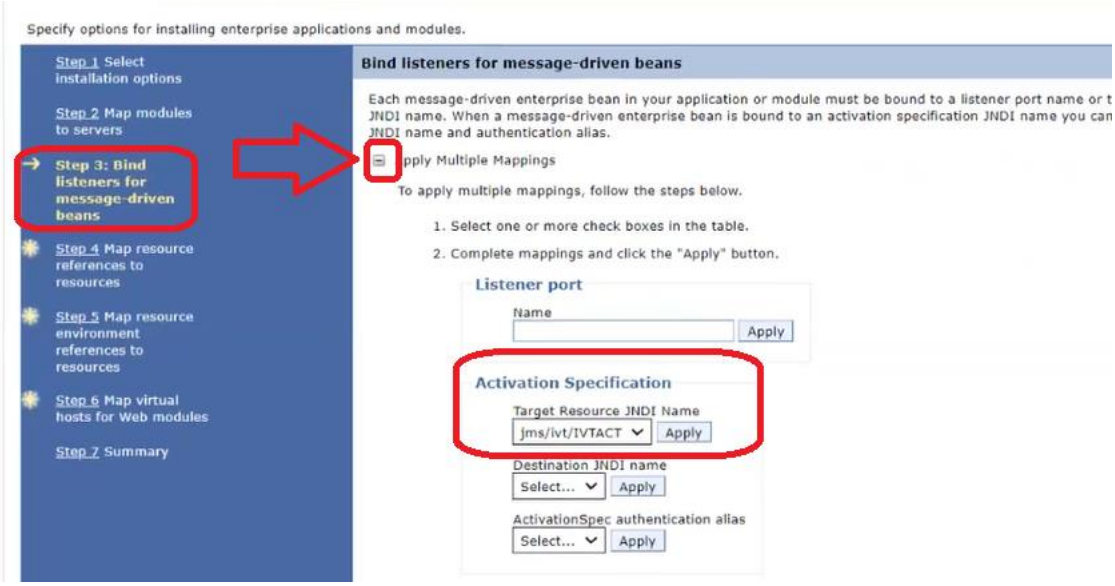
TRICKY STEP!

In "Step 3: Bind listeners for message-driven beans". You need to select you Activation Spec and perform an additional step on this page.

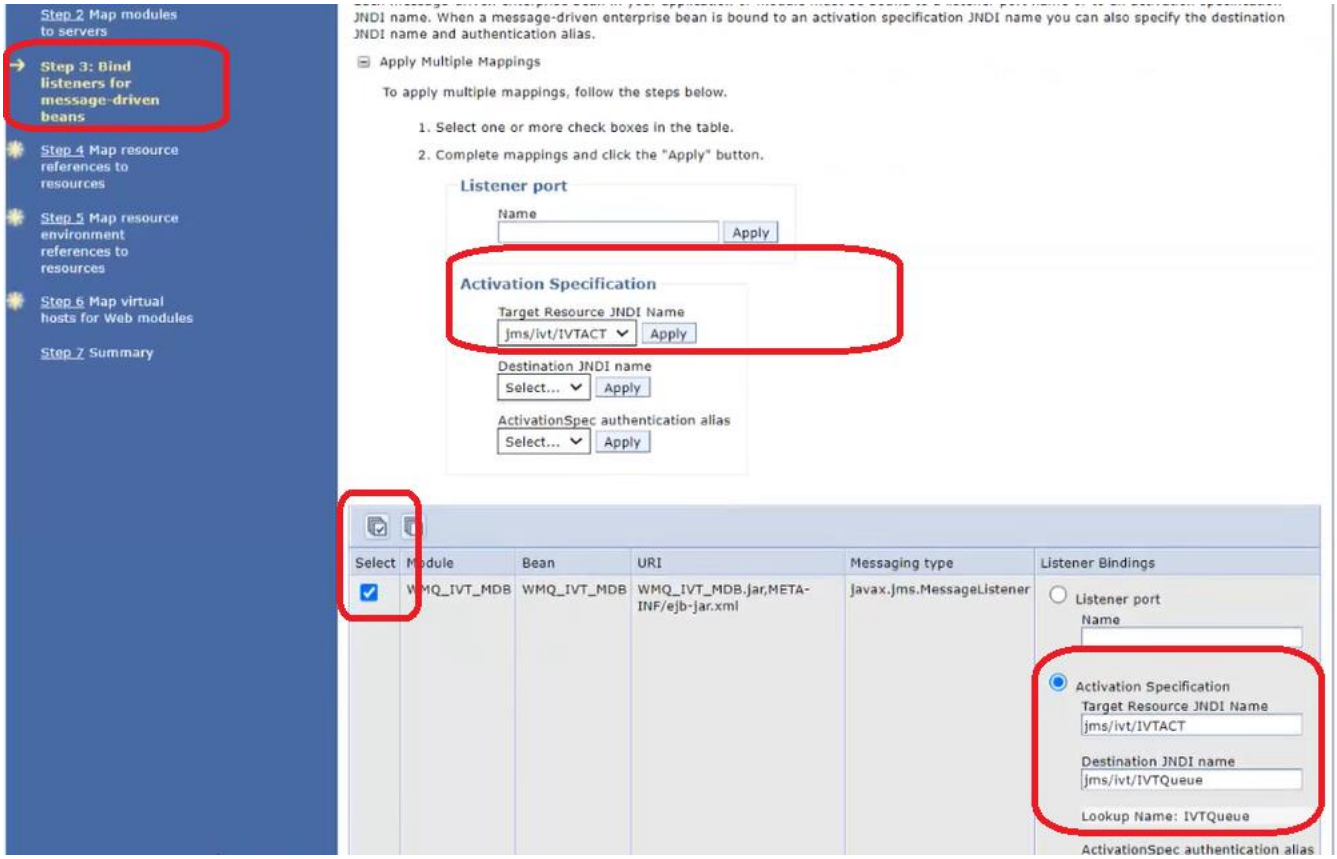


The tricky part is that you must click on:
Apply Multiple Mappings

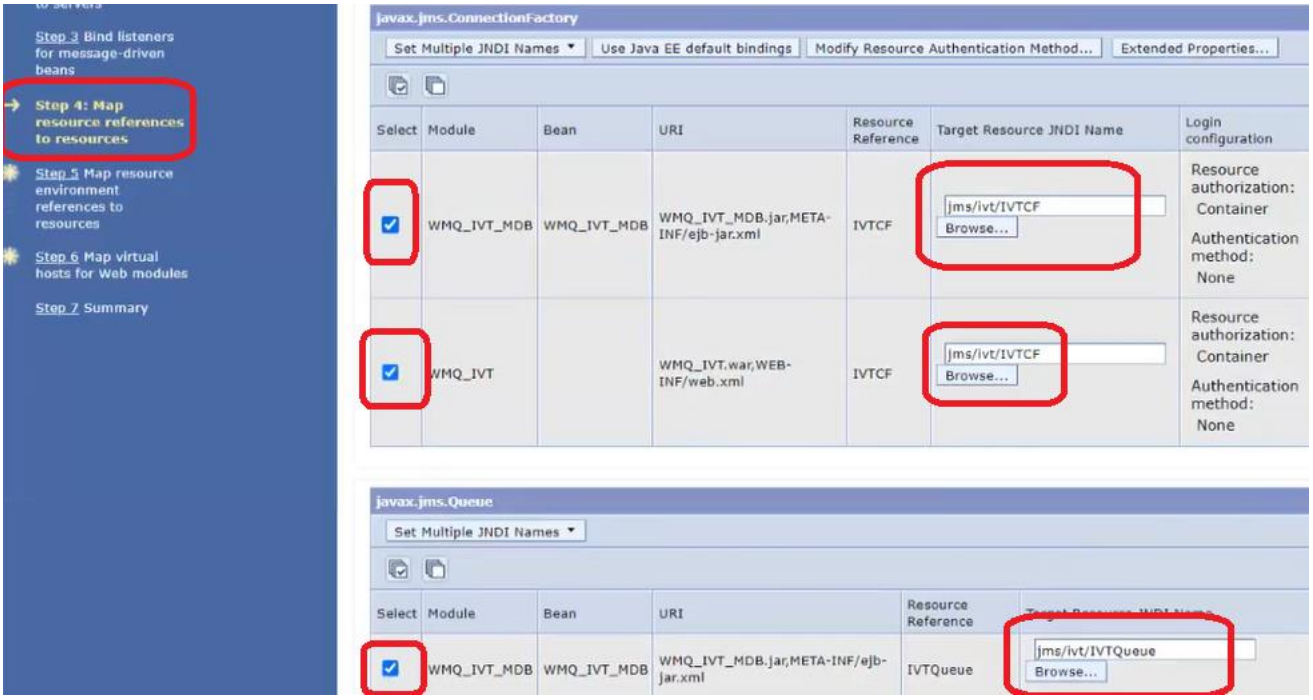
… and specify the Activation Specification



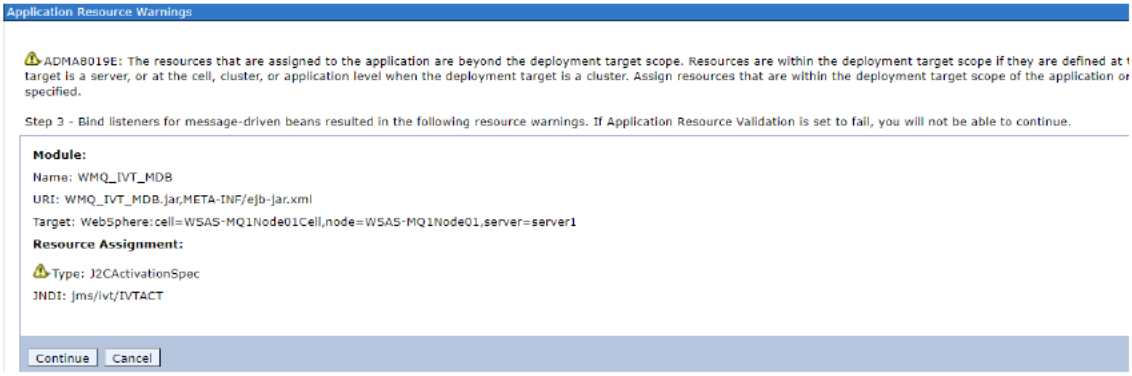
Again, you need to ensure that you specify an Activation Specification and that you select the item in the bottom panel.



In "Step 4: Map resource references to resources" you need to ensure that the "select" boxes are enabled and that the column "Target Resource JNDI Name" is properly populated.

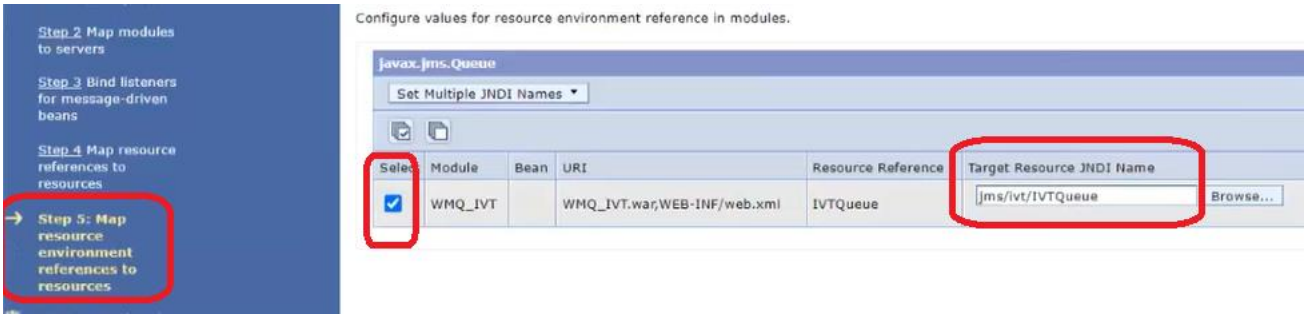


Attention:
If you get the following warning/error, you need to start again!

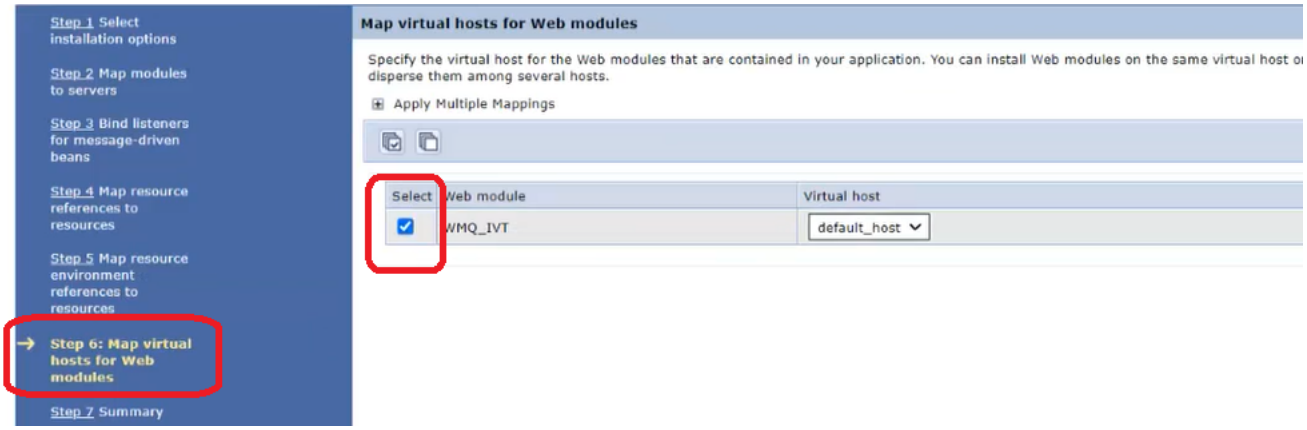


Let's assume that you did not get the above warning/error, then proceed with "Step 5: Map resource environment references to resources".

Ensure to Select the item and that the "Target Resource JNDI Name" is correct.



In "Step 6: Mao virtual hosts for Web modules" select the item.



In "Step 7: Summary" you will see something like this:

Step 1 Select installation options

Step 2 Map modules to servers

Step 3 Bind listeners for message-driven beans

Step 4 Map resource references to resources

Step 5 Map resource environment references to resources

Step 6 Map virtual hosts for Web modules

Step 7: Summary

Summary

Summary of installation options

Options	Values
Precompile JavaServer Pages files	No
Directory to install application	
Distribute application	Yes
Use Binary Configuration	No
Deploy enterprise beans	Yes
Application name	wmq.jmsra.ivt
Create MBeans for resources	Yes
Override class reloading settings for Web and EJB modules	No
Reload interval in seconds	
Deploy Web services	No
Validate Input off/warn/fail	warn
Process embedded configuration	No
File Permission	.*\.dll=755#.*\.so=755#.*\.a=755#.*\.sl=755
Application Build ID	9.3.0.0-p930-L220606
Allow dispatching includes to remote resources	No
Allow servicing includes from remote resources	No
Business level application name	
Asynchronous Request Dispatch Type	Disabled
Allow EJB reference targets to resolve automatically	No
Deploy client modules	No
Client deployment mode	Isolated
Validate schema	No
Cell/Node/Server	Click here

Click "Finish".

You will see the progress status…

Installing...

If there are enterprise beans in the application, the EJB deployment process can take several minutes. Do not save the configuration until the process completes.

Check the SystemOut.log on the deployment manager or server where the application is deployed for specific information about the EJB deployment process as it occurs.

ADMA5016I: Installation of wmq.jmsra.ivt started.

ADMA5067I: Resource validation for application wmq.jmsra.ivt completed successfully.

ADMA5058I: Application and module versions are validated with versions of deployment targets.

ADMA5018I: The EJBDeploy program is running on file /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/wstemp/3500252/upload/wmq.jmsra.ivt.ear

Starting workbench.

EJB Deploy configuration directory: /opt/ Please Wait... er/profiles/AppSrv01/ejbdeploy/configuration/

framework search path: /opt/IBM/WebSp /itp/plugins

build:RADWEJB95-I20150829_0214

You need to ensure to click on "Save" at the end:

0 Errors, 0 Warnings, 0 Informational Messages

ADMA5007I: The EJBDeploy program completed on file /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/wstemp/3500252/upload/wmq.jmsra.ivt.ear

ADMA5005I: The application wmq.jmsra.ivt is configured in the WebSphere Application Server repository.

ADMA5005I: The application wmq.jmsra.ivt is configured in the WebSphere Application Server repository.

ADMA5081I: The bootstrap address for client module is configured in the WebSphere Application Server repository.

ADMA5053I: The library references for the installed optional package are created.

ADMA5005I: The application wmq.jmsra.ivt is configured in the WebSphere Application Server repository.

ADMA5001I: The application binaries are saved in /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/wstemp/3500252/workspace/cells/WSAS-MQ1Node01Cell/applications/wmq.jmsra.ivt.ear/wmq.jmsra.ivt.ear

ADMA5005I: The application wmq.jmsra.ivt is configured in the WebSphere Application Server repository.

ASEC0400I: Successfully updated the application wmq.jmsra.ivt with the appContextIDForSecurity information.

ADMA5005I: The application wmq.jmsra.ivt is configured in the WebSphere Application Server repository.

ADMA5005I: The application wmq.jmsra.ivt is configured in the WebSphere Application Server repository.

ADMA5113I: Activation plan created successfully.

ADMA5011I: The cleanup of the temp directory for application wmq.jmsra.ivt is complete.

ADMA5013I: Application wmq.jmsra.ivt installed successfully.

To start the application, first save changes to the master configuration.

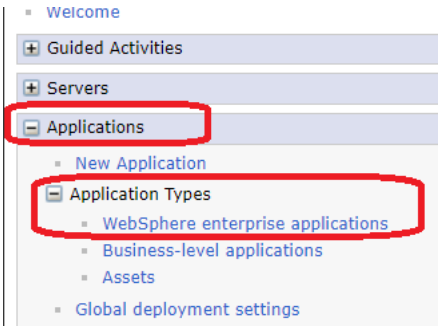
Changes have been made to your local configuration. You can:

•Save directly to the master configuration.

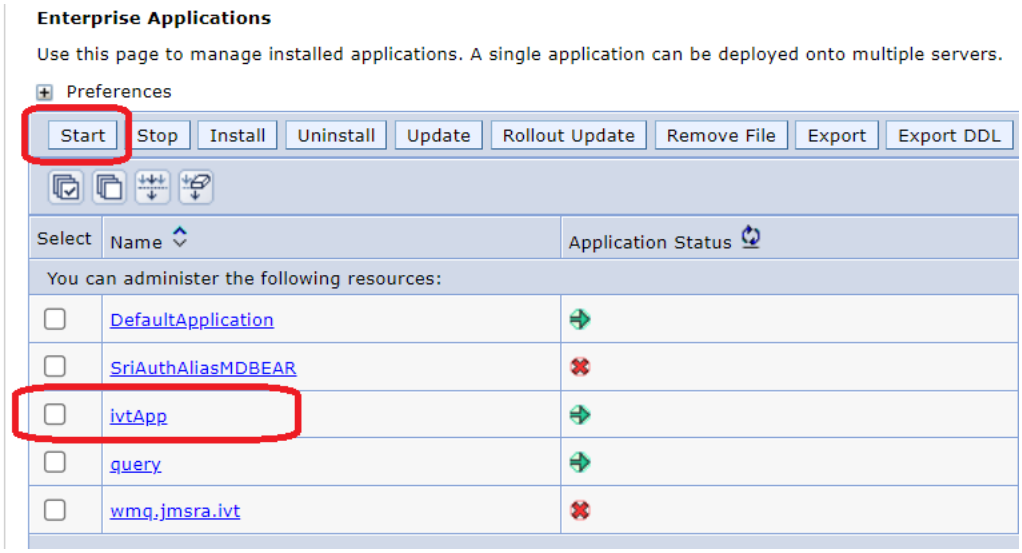
Now that the IVT application has been deployed, you can start it.

In the left navigation panel, select:

Applications > Application Types > WebSphere enterprise applications



In the right navigation panel, if "ivtApp" is not running (green arrow in column "Application Status"), then you can select it and click on the button "Start".



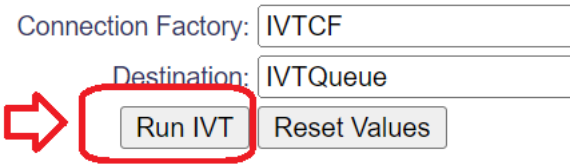
+ Open a Web Browser and specify the URL for the IVT

Using the Microsoft Edge browser from a Windows PC:

http://wsas-mq1.fyre.ibm.com:9080/WMQ_IVT/

You will see the following:

Now click on the button:



And you should see:

IBM MQ JavaEE 7 Connector Architecture IVT

Sat Aug 19 08:25:21 PDT 2023

Running Installation Verification Test:

Using Connection Factory:*IVTCF*
Using Destination:*IVTQueue*

Creating initial context...	✓
Looking up MQ Connection Factory...	✓
Looking up Destination...	✓
Creating connection...	✓
Starting connection...	✓
Creating session...	✓
Creating a temporary reply queue...	✓
Creating message consumer...	✓
Creating message producer...	✓
Creating message...	✓
Sending message to the MDB...	✓
Receiving response message from the MDB...	✓
Closing connection...	✓

Installation Verification Test completed successfully!

View Message Contents

Re-run Installation Verification Test

YEAH!!!

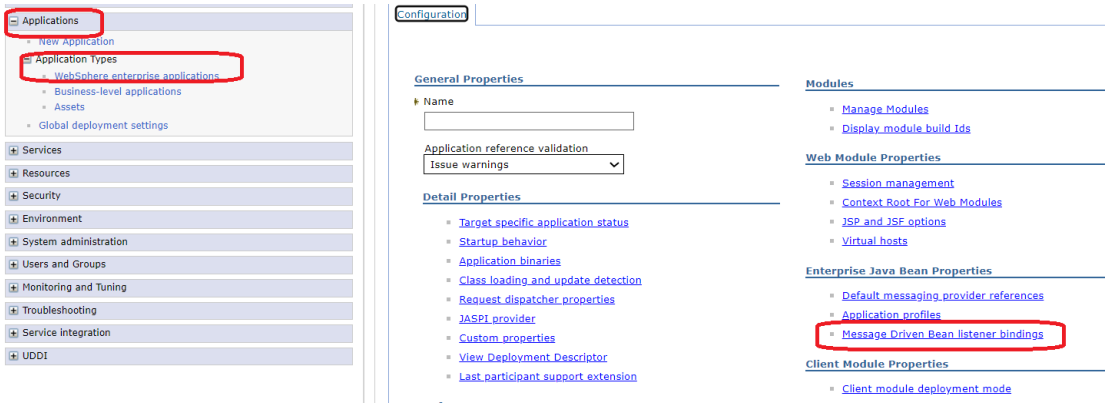
+ (If necessary) Troubleshooting

What if the application does not start?

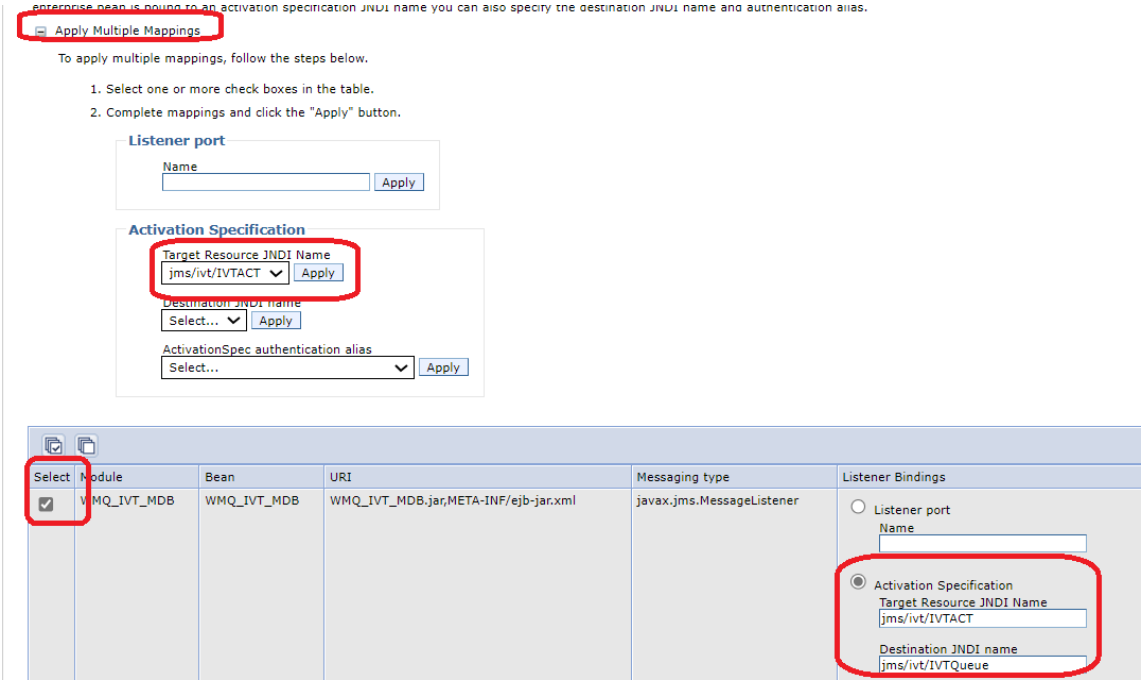
1) Check the SystemOut.log of the WSAS server for errors or warnings:

/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/logs/server1/SystemOut.log

2) From the WSAS Console, try clicking on the App, then click on “Message Driven Bean Listener Bindings”.



This will take you back the bindings dialog screens, where you can redo the “apply” for the Activation Spec and make sure the box is checked.



+++ end +++