PeopleSoft

EnterpriseOne Xe Adapter for MQSeries – Configuration Guide (UNIXv2) PeopleBook

June 2000

J.D. Edwards World Source Company
7601 Technology Way
Denver, CO 80237

SKU XeEAUN

Table of Contents

Introduction

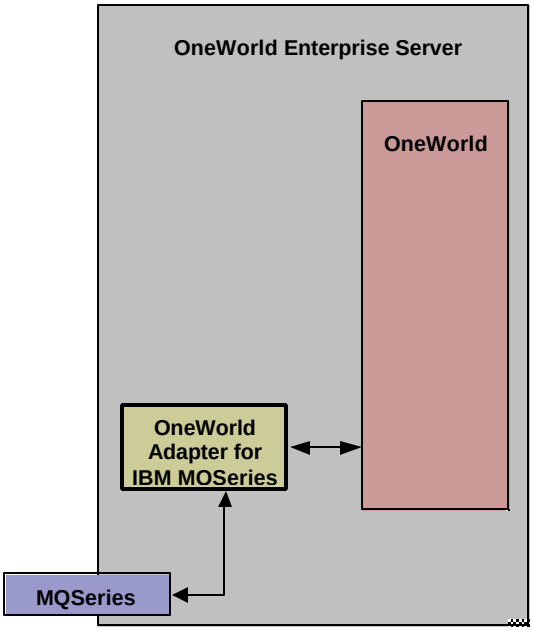
The OneWorld Adapter for MQSeries allows you to connect any e-storefront or other application to OneWorld by sending and receiving messages through IBM's MQSeries messaging system. The adapter monitors an inbound queue for Request/Reply messages, performs the requested services, and places the results on outbound queues. It also monitors OneWorld for certain activities and publishes the results in an outbound message queue. All messages transported through MQSeries are in the form of XML documents.

The purpose of this document is to make you successful with the configuration and operation of the adapter. There is a separate document available for a systems developer that explains the design of the adapter, the formats of the documents, and those tasks necessary to create, modify, and process the XML documents.

This adapter was originally developed to provide a connection between OneWorld and the IBM storefront products WebSphere Commerce Suite and Commerce Integrator. However, the adapter design is such that it avoids any program-level dependencies on the Commerce Suite and Commerce Integrator suite of products. The OneWorld Adapter for MQSeries is a separate product from that suite that can be licensed and installed independently. It can be used to connect OneWorld with any front end that can implement the IBM MQSeries messaging protocols, and provide/consume XML documents in the prescribed formats. The OneWorld Adapter for MQSeries is simply designed to export and import XML documents through MQSeries in the prescribed formats.

The following illustration shows the OneWorld Enterprise Server with the MQSeries adapter included.

Diagram: OneWorld Adapter for MQSeries on the OneWorld Enterprise Server



About This Document

Due to the ever-changing nature of Web products in general, the content design is single sourced where practical and possible. This means that the document is primarily designed for "online" use. However, some format deliverables are better suited for online delivery than others.

The architecture of this document is designed to directly present you with only the information you most likely need to perform an installation of the J.D. Edwards Storefront product. However, additional information is presented in the form URLs and embedded documents in the form of Word documents and PDF documents.

Your J.D. Edwards Storefront CD should contain two forms of these Installation Instructions:

- **Microsoft Word source document.** Since this is the source application in which the document was authored, it is the most fully featured of the delivered document structures. All hyperlinks are active and all embedded documents (either Word or PDF) are directly accessible by double-clicking on the icons.
- **PDF version of the Word source document.** In this version of the document, the embedded URLs, hyperlinks, and documents are visible but not live or accessible by clicking. For URLs, you will need to copy them from Acrobat and into your browser. For hyperlinks, there is no workaround other than using the Find function within Adobe Acrobat. For embedded documents, you will need to note the document name and manual locate and open the document. These embedded document attachments are included as separate files in the root of the J.D. Edwards Storefront CD.

STEP 1: Set Up MQSeries Queues

MQSeries is a queue messaging system that requires a sender and receiver relationship necessary for queue communications. One side of MQSeries is installed on the OneWorld Enterprise Server while the related side is installed on another physical or logical machine. The setup on the OneWorld server enables OneWorld to receive inbound messages from an e-storefront application and also to place outbound messages into a queue for processing by an e-storefront application. Refer to the applicable IBM documentation for instructions on installing MQSeries on machines other than Commerce Integrator Servers.

The procedures described in this section assume you have already installed the IBM product MQSeries onto your OneWorld Enterprise Server.

This section involves setting up MQSeries on the following machines:

- OneWorld MQSeries Server (inbound to OneWorld and outbound to storefront)
- Non-OneWorld MQSeries Server (inbound to storefront and outbound to OneWorld)
- Communications Between the OneWorld MQSeries Server and the Non-OneWorld MQSeries Server
- Working with MQSeries Queue Manager commands

Before You Begin

On your RS6000 system, you must add MQM to the group for user JDE.

OneWorld MQSeries Server (inbound to OneWorld and outbound to storefront)

This section describes the following MQSeries functions that you must perform on the OneWorld Enterprise Server:

- Creating MQSeries Queue Manager
- Creating MQ local queues
- Creating a MQ remote queue
- Creating a MQ local sender channel
- Creating a MQ local receiver channel
- Working with MQSeries Queue Manager commands

? To create and start MQSeries Queue Manager

Tips and Techniques
The names of queues and channels in MQSeries are case-sensitive. Be sure and use all caps as specified in this document. You can specify any name for a queue or channel. However, it is important that the queue names you create match the queue names you specify in the jde.ini file on the Enterprise Server.

On your OneWorld Enterprise Server, open a shell:

1. Execute the following command to create the Queue Manager:

   ```
   crtmqm -c "JD Edwards queue manager" JDE_QMGR
   ```

   This establishes a queue manager that is not the default queue manager.

2. Once that is created you must start the queue manager using the following command:

   ```
   strmqm JDE_QMGR
   ```

? To create MQ local queues

Tips and Techniques
The names of queues and channels in MQSeries are case-sensitive. Be sure and use all caps as specified in this document. You can specify any name for a queue or channel. However, it is important that the queue names you create match the queue names you specify in the jde.ini file on the Enterprise Server.

First you must start the MQSC facility interactively. In the shell you used to start the queue manager, enter the following command:

```
runmqsc JDE_QMGR
```

1. You are now ready to define the following local queues:
   - INBOUND.Q
   - SUCCESS.Q
   - ERROR.Q
   - DEFRES.Q
   - OUTBOUND.Q.XMIT
2. Run the following MQSeries commands to create the above local queues:

```
Define QLOCAL(INBOUND.Q) DEFPSIST(YES)
DESCR('Queue for message into OneWorld')

Define QLOCAL(SUCCESS.Q) DEFPSIST(YES)
DESCR('Queue for successful message in OneWorld')

Define QLOCAL(ERROR.Q) DEFPSIST(YES)
DESCR('Queue for error messages in One World')

Define QLOCAL(DEFRES.Q) DEFPSIST(YES)
DESCR('Queue for responses if not in message in One World')

Define QLOCAL(OUTBOUND.Q.XMIT) DEFPSIST(YES)
USAGE(XMITQ) DESCR('Transmit queue to NC system')
```

? To create the MQ remote queue

Tips and Techniques

The names of queues and channels in MQSeries are case-sensitive. Be sure and use all caps as specified in this document. You can specify any name for a queue or channel. However, it is important that the queue names you create match the queue names you specify in the jde.ini file on the Enterprise Server.

In the same shell you used in the previous task, enter the following command to create a single MQ remote queue:

```
Define QREMOTE(OUTBOUND.Q) DEFPSIST(YES)
XMITQ(OUTBOUND.Q.XMIT) RNAME(ECE_IN2MQI)
RQMNAME(ECE_MQI_QMGR) DESCR('One World out bound queue to NC system')
```

? To create a MQ local sender channel

Tips and Techniques

The names of queues and channels in MQSeries are case-sensitive. Be sure and use all caps as specified in this document. You can specify any name for a queue or channel.

In the same shell you used in the previous task, enter the following command to create a MQ local sender channel:

```
Define CHANNEL(OW2MQI_CHL) CHLTYPE(SDR) TRPTYPE(TCP)
CONNAME(Net Comm Server Name) XMITQ(OUTBOUND.Q.XMIT)
DISCINT(0) DESCR('Sender channel to NC system')
```

? To create a MQ local receiver channel

Tips and Techniques

The names of queues and channels in MQSeries are case-sensitive. Be sure and use all caps as specified in this document. You can specify any name for a queue or channel.

In the same shell you used in the previous task, enter the following command to create a MQ local receiver channel:

```
Define CHANNEL('MQI2OW_CHL') CHLTYPE(RCVR)
TRPTYPE(TCP)  DESCR('Receiver channel from WCS system')
```

? To end the MQSeries Queue Manager

When you are finished creating the queues and channels, from the same shell you used in the previous tasks, enter the following command:

```
END
```

This exits the runmqsc program.

Non-OneWorld MQSeries Server (inbound to OneWorld and outbound to storefront)

MQSeries is a queue messaging system that requires a sender and receiver relationship necessary for queue communications. One side of MQSeries is installed on the OneWorld Enterprise Server while the related side is installed on another physical or logical machine, in this case called the non-OneWorld MQSeries Server.

This section contains procedures to create and maintain the queues required to communicate with the MQSeries server running on the OneWorld Enterprise Server. If MQSeries queues on the non-OneWorld MQSeries are on a machine containing the IBM Commerce Integrator (ICI), the operating system for that machine must be NT, AIX, or Solaris. J.D. Edwards has supplied basic queue setup instructions for each OS in the *WCS Integration Reference* document. That document is supplied in the following directory on the CD containing the OneWorld Adapter for MQSeries:

```
\Documentation\WCS_Integration_Reference
```

Because the sequence of installing the queues and establishing communications between the queues is tightly linked, you should immediately return to the next step in this document after you have created the queues on the non-OneWorld or ICI machine.

Tips and Techniques
After you install and configure MQSeries queues, you must re-boot the machine in order to set the ownership for system variables established during the process.

Communications Between the OneWorld MQSeries Server and the Non-OneWorld MQSeries Server

You must set up communications between the OneWorld MQSeries Server (functioning on the OneWorld Enterprise Server) and the Non-OneWorld MQSeries Server using the sequenced and machine-dependent steps in this section.

Note: Testing in laboratory environments presents an anomaly which may be applicable to field installations. It was noted that when channels were started for the first time, it was necessary for both the sender and receiver channels to be manually started on the local and remote machines, respectively. However, subsequent to the initial start of the channel, it was only necessary to start the sender on the local machine. In turn, that caused the receiver on the remote machine to automatically start.

- Start the Queue Manager on the non-OneWorld MQSeries Server
- Start the Queue Manager on the OneWorld MQSeries Server
- Start the channel on the non-OneWorld MQSeries Server
- Start the channel on the OneWorld MQSeries Server

? To start the Queue Manager on the non-OneWorld MQSeries Server

On the non-OneWorld MQSeries Server:

1. In a shell, start the Queue Manager by entering the following command:

   ```
   strmqm ECE_MQI_QMGR
   ```

2. In the same shell, start the listener by entering the following commands:

   ```
   runmqlsr -m JDE_QMGR -t TCP
   ```

   where this is an interactive command that ends when terminal session ends.

   ```
   nohup runmqlsr -m JDE_QMGR -t TCP &
   ```

   where this is set equivalent to batch process and defines no hang up; this is run as background process.

? To start the Queue Manager on the OneWorld MQSeries Server

On the OneWorld MQSeries Server:

3. In a shell, start the Queue Manager by entering the following command:

   ```
   strmqm ECE_MQI_QMGR
   ```

4. In the same shell, start the listener by entering the following commands:

   ```
   runmqlsr -m JDE_QMGR -t TCP
   ```

   where this is an interactive command that ends when terminal session ends.

   ```
   nohup runmqlsr -m JDE_QMGR -t TCP &
   ```

   where this is set equivalent to batch process and defines no hang up; this is run as background process.

? To start channel on the non-OneWorld MQSeries Server

On the non-OneWorld MQSeries Server

In the same shell as used in the previous task, enter the following command:

```
runmqchl MQI2OW_CHL
runmqchl OW2MQI_CHL
```

Note: Testing in laboratory environments presents an anomaly which may be applicable to field installations. It was noted that when channels were started for the first time, it was necessary for both the sender and receiver channels to be manually started on the local and remote machines, respectively. However, subsequent to the initial start of the channel, it was only necessary to start the sender on the local machine. In turn, that caused the receiver on the remote machine to automatically start.

? To start the channel on the OneWorld MQSeries Server

On the OneWorld MQSeries Server:

In the same shell as used in the previous task, enter the following command:

```
runmqchl OW2MQI_CHL
runmqchl MQI2OW_CHL
```

Note: Testing in laboratory environments presents an anomaly which may be applicable to field installations. It was noted that when channels were started for the first time, it was necessary for both the sender and receiver channels to be manually started on the local and remote machines, respectively. However, subsequent to the initial start of the channel, it was only necessary to start the sender on the local machine. In turn, that caused the receiver on the remote machine to automatically start.

Working with MQSeries Queue Manager commands

Tips and Techniques
The names of queues and channels in MQSeries are case-sensitive. Be sure and use all caps as specified in this document. You can specify any name for a queue or channel. However, it is important that the queue names you create match the queue names you specify in the jde.ini file on the Enterprise Server.

The following is a list of commands you can use when working with MQSeries Queue Manager commands.

1. Start the Queue Manager by issuing the following command:

   ```
   strmqm JDE_QMGR
   ```

2. End the Queue Manager by issuing the following command where the –i switch means immediately:

   ```
   endmqm –i JDE_QMGR
   ```

3. Run the following command to start using MQSeries commands:

   ```
   runmqsc JDE_QMGR
   ```

4. Display the local queue by running the following command:

   ```
   display QL(qname.Q)
   ```

5. Clear the local queue by running the following command:

   ```
   clear QL(qname.Q)
   ```

6. Enter the following command to stop using MQSeries commands:

   ```
   END
   ```

7. Run the Browse Queue command. This does not remove the message.

   ```
   amqsbcg qname.Q JDE_QMGR
   ```

8. To start the channel, issue the following command:

   ```
   runmqchl(channel_name)
   ```

9. Start a listener by issuing the following command

   ```
   runmqslr <Gateway Name> TpName JDE_QMGR
   ```

   where *<Gateway Name>* is the gateway name of the machine on which the listener is being run.

10. If a channel goes inactive, you can reset it by issuing the following command:

    ```
    resetchl (channel_name)
    ```

STEP 2: Set Up OneWorld UBE Queues

You must set up at least two OneWorld UBE queues. This is required because the OneWorld Adapter for MQSeries relies on the use of a subsystem UBE (R00460). Like all subsystem UBEs, the R00460 operates in a permanent processing mode that consumes the queue in which it runs to an extent where no other UBE jobs running on the server can operate in the same queue. As a result, you must define at least two UBE queues where one queue is dedicated to normal UBE processing and the other is dedicated to the OneWorld Adapter for MQSeries subsystem.

This section describes:

- Modifying the jde.ini on the OneWorld Enterprise Server to support multiple UBE queues
- Modifying the jde.ini on a OneWorld client to submit the UBE subsystem request

Additional platform-specific information on working with queues is available in *the OneWorld Server and Workstation Administration Guide.*

? To modify the jde.ini on the OneWorld Enterprise Server to support multiple UBE queues

Ensure the following settings are correct:

Parameter	Description

`[NETWORK QUEUE SETTINGS]`
`UBEQueues=2`
`UBEQueue1=ONEWORLD`
`UBEQueue2=QBATCH`
	The following values should be added or modified where:
	`UBEQueues=` Defines the number of UBE queues that are defined on this OneWorld Enterprise Server. In this example, two queues are defined.
	`UBEQueue1=` Defines the name of the first UBE queue. In this example, the queue name is ONEWORLD.
	`UBEQueue1=` Defines the name of the second UBE queue. In this example, the queue name is QBATCH.

? To modify the jde.ini on a OneWorld client to submit the UBE subsystem request

If you submit or start the R00460 subsystem UBE from a OneWorld Windows client, you must first temporarily modify that client's jde.ini file. The temporary modification is required so that the client can direct the appropriate UBE queue name. After the client submits or starts the R00460 subsystem UBE, you must undo the temporary change so that client can regain access to normal UBE submissions to the server-based UBE queue.

On the client machine from which you want to submit or start the R00460 subsystem UBE, ensure the following jde.ini settings are correct:

Parameter	Description	
[NETWORK QUEUE SETTINGS] UBEQueue=	UBEQueue=	Defines the name of the local or server-based UBE queue. If you want the client to start or submit the R00460 subsystem, you should enter a value that corresponds with the value set by the UBEQueue1= on the OneWorld Enterprise Server. In the example presented in this guide, the name of the R00460 subsystem queue is ONEWORLD. If you want the client to use the normal server-based UBE processing queue, you should enter a value that corresponds with the equivalent value for that queue on the Enterprise Server. In the example presented in this guide, the name of the normal UBE processing queue is QBATCH.

Tips and Techniques
Any time you modify settings in the jde.ini file on the client, you must exit and restart OneWorld in order for those changes to become effective.

STEP 3: Configure the OneWorld Enterprise Server

You must configure the Enterprise Server with the settings as described in this section. Perform the following tasks:

- Configure the jde.ini file on the OneWorld Enterprise Server
- Configure the .oneworld script on the OneWorld Enterprise Server

Tips and Techniques
Make sure that you have not brought the NT Services back up until after you make the requisite changes to the jde.ini file and the .oneworld script on the OneWorld Enterprise Server.

? To configure the jde.ini file on the OneWorld Enterprise Server

Ensure the following sections and settings are correct.

Parameter	Description

`[SECURITY]`
`User=JDE`
`Password=JDE`
	You must have these values set in order for OneWorld to accept the outbound messages from WebSphere Commerce Suite.

`[JDENET]` `maxKernelRanges=13`	This value represents the total number of JDENET kernel types. When the MQ kernel_def type is added, you must increment this value by one so that JDENET knows to include and startup this added kernel.

`[JDENET_KERNEL_DEF13]`
`krnlName=MQSI Kernel`
`beginningMsgTypeRange=5513`
`endingMsgTypeRange=6001`
`dispatchDLLName=libmqsadapt.so`
`dispatchDLLFunction=`
`JDEK_DispatchMQSeriesProcess`
`maxNumberOfProcesses=1`
`numberOfAutoStartProcesses=1`
	You must update the “DEF” value to next number. For B733.1 and B733.2, the value is 13. For B733.3, the value is 14.
	Valid value for the parameter, numberOfAutoStartProcesses are:
	0 = The MQSI kernel will not automatically start when OneWorld is initialized.
	1 = The MSQI kernel will always be automatically started when OneWorld is initialized. You must use this setting to use the OneWorld Adapter for MQSeries.

Parameter	Description
[MQSI] QMGRName=JDE_QMGR QInboundName=INBOUND.Q QErrorName=DEFRES.Q QOutboundName=OUTBOUND.Q TimeoutWaitInterval=15 MaxBufferLength=10240 CreateHeader=YES AppGroup=NNJDE JDEOrderStatusCode=JDESOOUT JDECustomerCode=JDEAB JDEItemPriceCode=JDEPRICE JDEItemQtyCode=JDEIL NCOrderStatusCode=JDE.IC.F4201Z1 NCCustomerCode=JDE.IC.F0101Z2 NCProductPriceCode=JDE.IC.F4106NC NCProductQtyCode=JDE.IC.F41021Z1 OWHostName=*host_name*	These settings are for the header information on the message that is required for Commerce Integrator. The name of the queues can be any name, but must match the names you specify in the MQSeries queue setup. If the adapter is being used without WebSphere Commerce Integrator, you can specify the create header as equal to No. Also in that case, you should set those subsequent Commerce Integrator-specific parameter settings in the MQSI section to blank. This includes the four parameters prefixed by JDE and four parameters prefixed by NC. In either case, you must specify a *host_name* for the OWHostName= parameter. The name you specify here is the machine used to create the net message to trigger the Outbound Adapter. This is the server on which OneWorld resides.

? To configure the .oneworld script on the OneWorld Enterprise Server

You must edit the UNIX script file called .oneworld that is located in the $HOME directory. This script is generated by the initial OneWorld server installation program. It is used by the UNIX system to read and set the OneWorld environment variables. You must edit the script to add an environment variable for the OneWorld Adapter for MQSeries.

1. Shut down OneWorld before you edit the .oneworld script file.
2. Locate the .oneworld script file in the $HOME directory.
3. Open the .oneworld script file by issuing the following command:

   ```
   vi $HOME/.oneworld
   ```

4. Add the following line command to the .oneworld script:

   ```
   export ICU_DATA=$SYSTEM/locale/xml/
   ```

5. Save the close the .oneworld script file.
6. Log out of your current user session.
7. Log back onto your user session to enable the OneWorld Adapter for MQSeries environment variable.

STEP 4: Configure OneWorld to Enable Transaction-Specific Outbound Processing

Transaction-specific outbound processing uses Z-Table processes supported by OneWorld applications. The outbound processes available to you vary depending on which OneWorld release you use. Additional transaction-specific outbound processes might be available in a future ASU or ESU.

If you use transaction-specific outbound processing you must perform configuration tasks such as setting up processing options or setting up data export controls.

This section contains those tasks that you must perform on the OneWorld applications to enable transaction-specific outbound processing. These tasks include:

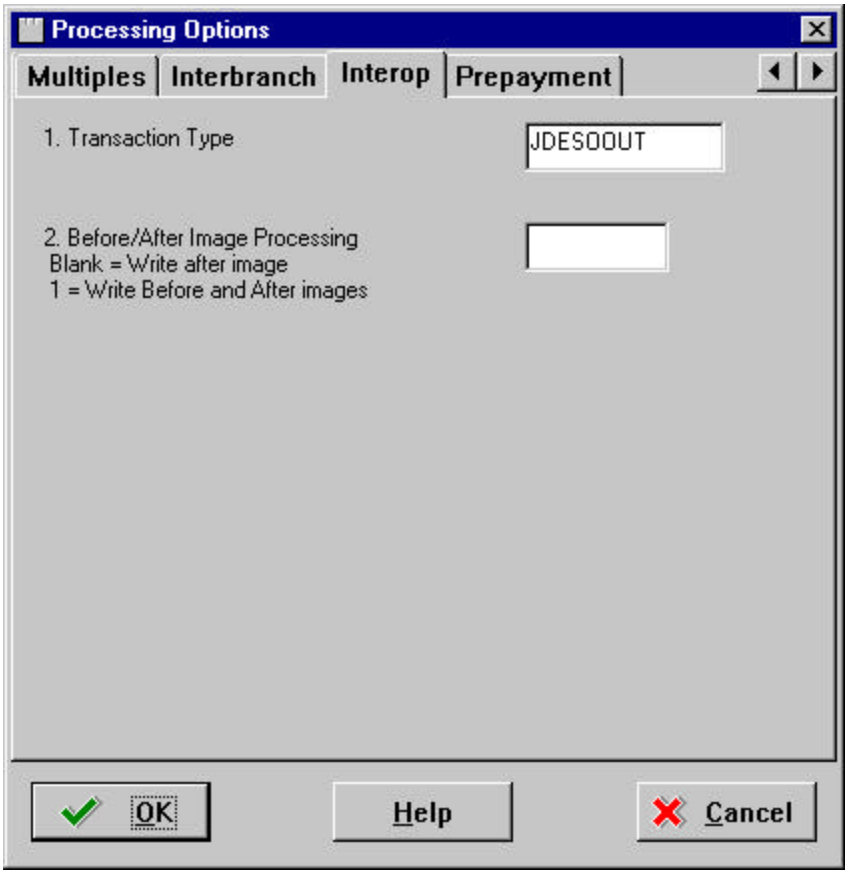
- Setting up the Processing Options for the Sales Order Master Business Function (P4210)
- Setting up the Processing Options for the Address Book Master Business Function (P0100041)
- Setting up the Processing Options for Address Book Revisions (P01012)
- Setting up Data Export Controls

? To setup the processing options for the Sales Order Entry Master Business Function (P4210)

You must setup the OneWorld Sales Order Entry program to enable it for interoperability operations.

1. From the client fast path to G4211.
2. Right click on Sales Order Detail (P4210).
3. Select Prompt For Values.
4. Click on the Interop tab.
5. Enter JDESOOUT. This defines the transaction type.

6. Enter a value for Option 2 to define the before/after image processing. Valid values are blank or 1.



7. Click OK.

? To setup the Processing Options for the Address Book Master Business Function (P0100041)

You must setup the Address Book Master Business Function to enable interoperability operations.

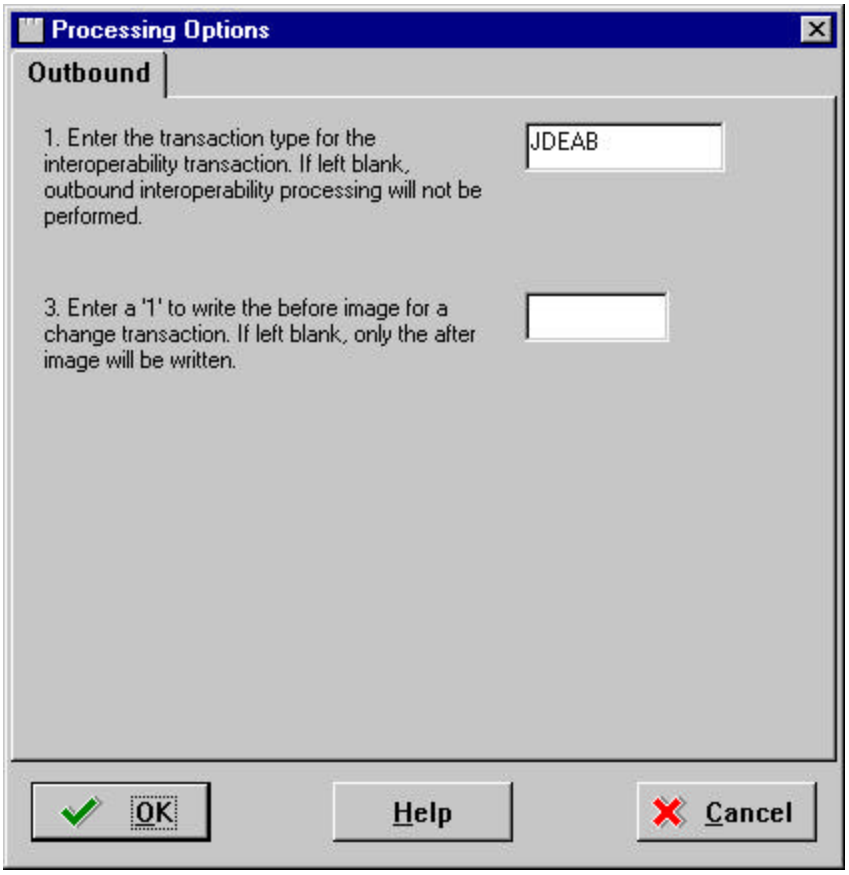
1. Fast path to GH9011.
2. Select Interactive Version.
3. Enter P0100041 to the Interactive Application and click Find.
4. If there is not a Version called INTOP select version ZJDE0001 and copy it.
5. Call the new version INTOP and click OK.

6. Highlight the version INTOP and click on Processing Options in the Exit Bar.

7. Select the Outbound tab.

8. For Option 1, enter JDEAB.

9. Enter a value for Option 3 to define the before/after image processing. Valid values are blank or 1.



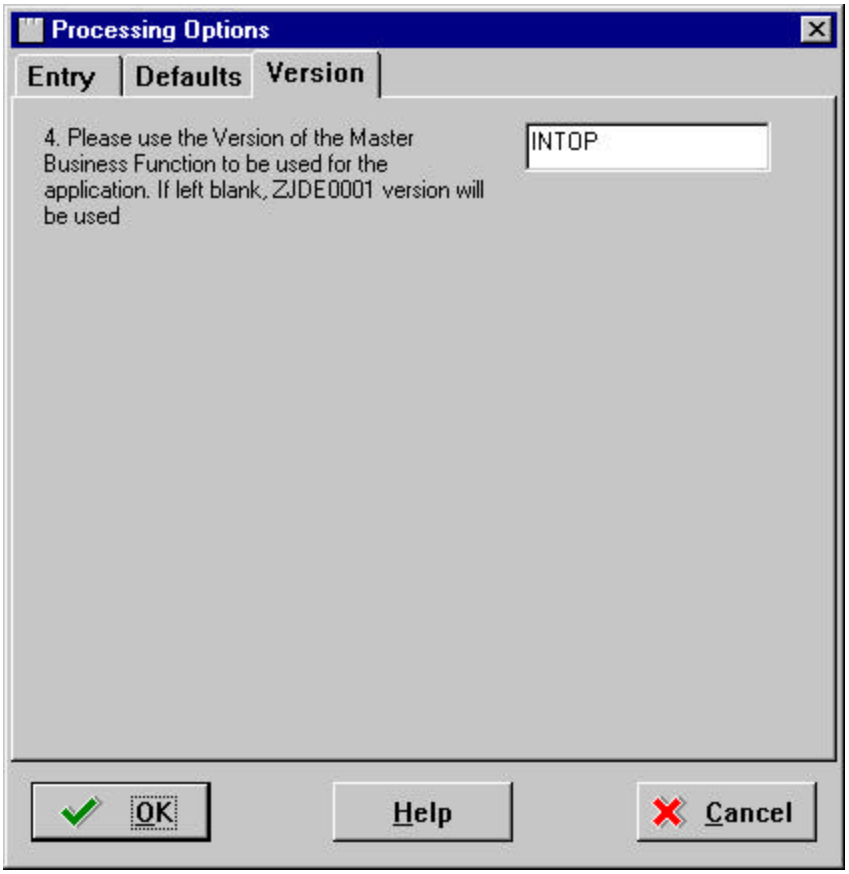
10. Click OK.

11. Click Close.

? To setup the Processing Options for Address Book Revisions (P01012)

You must setup the Address Book Revisions to enable interoperability operations.

1. Fast path to G01.
2. Right click on Address book Revisions (P01012).
3. Select Prompt For Values.
4. Click on the Version tab.
5. For Option 4, enter a value of INTOP.

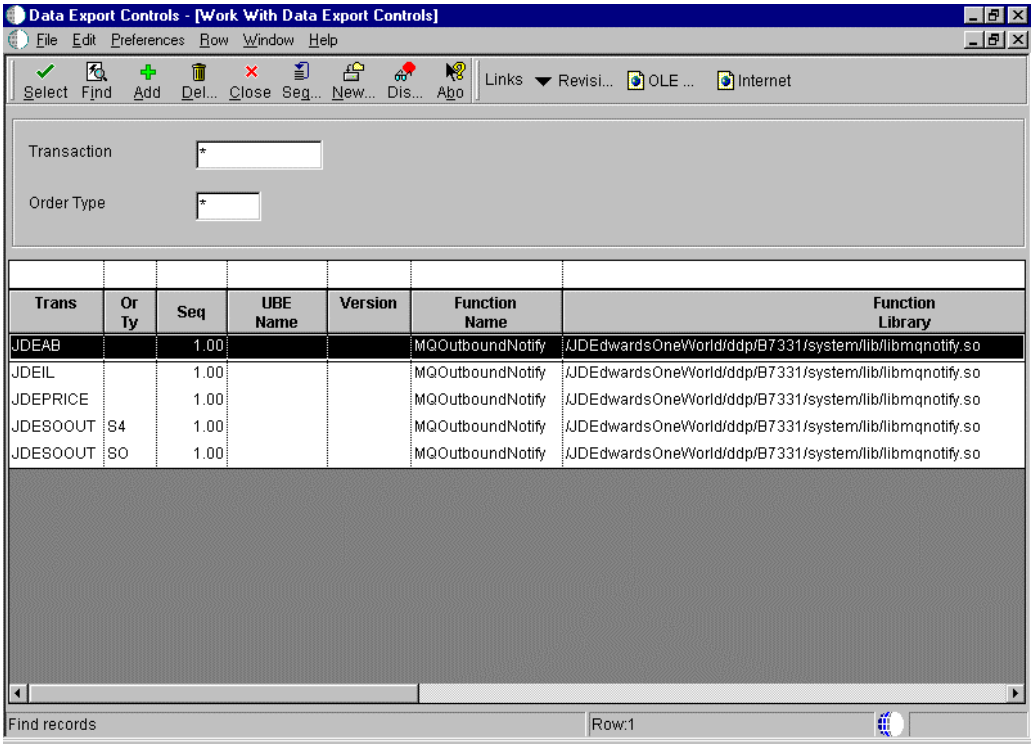


6. Click OK.
7. Click Close.
8. Click Close.

? Setting up Data Export Controls

You must insert a F0047 record for each transaction type. The record specifies the vendor-specific UBE or function to call to process the transaction. For transaction type JDESOOUT you must set up a record for each order type that you wish to export. See the *OneWorld B73.3.1 Interoperability Guide* for more information.

From menu G42A313, option P0047, add or verify the following data export controls:



Values for fields including those not shown in the diagram above include:

Form Field	Description
Trans	The values you specify in this field must match the values you have configured for OneWorld outbound processing in the various OneWorld applications. For the WCS integration, you must specify the following values: JDEAB JDEIL JDEPRICE JDESOOUT

Form Field	Description
Or Ty	The values you specify in this field must match the values that are required for your storefront. For example, for WCS the following order types are required: JDESOOUT S4 JDESOOUT SO
Seq	1.00
Function Name	MQOutboundNotify
Function Library:	/JDEdwardsOneWorld/ddp/B733/system/lib/libMQNotify.so The above location is shown as an example. The actual location will vary by individual site.
Execute for Add:	1
Execute for Update:	1
Execute for Delete:	1
Execute for Inquiry:	1
Flat File Export Mode:	0
External Database Export Mode	0
External API Export Mode:	0
Launch Immediately	1

STEP 5: Restart the UNIX Services

After you have completed the MQSeries queue setup and the various associated configuration tasks, you can restart the UNIX Services.

? To restart the UNIX Services

1. Start OneWorld Services for JDE B733 Network then JDE B733 Queue by entering the following commands:

   ```
   cd system/bin32
   ./RunOneWorld.sh
   ```

2. Successfully complete a porttest.

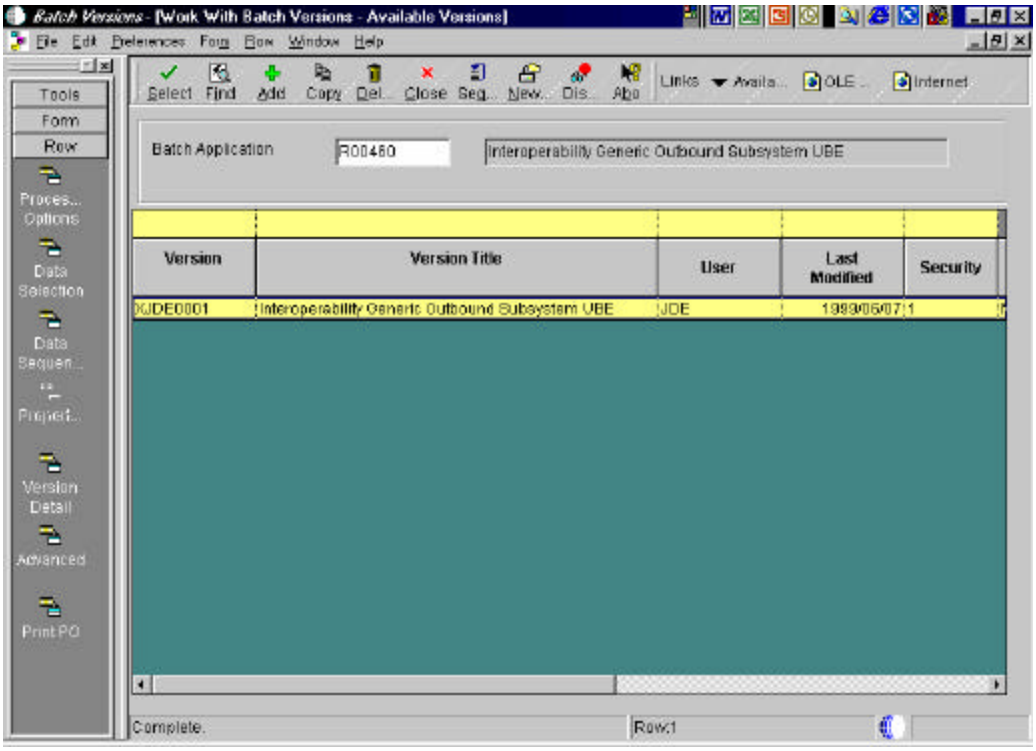
STEP 6: Start the OneWorld R00460 Subsystem for MQSeries

You must start the R00460 subsystem for MQSeries on the OneWorld Enterprise Server. When running, this is the subsystem that monitors the MQSeries queues for messages and manages the OneWorld side of the queues.

? To start the R00460 subsystem

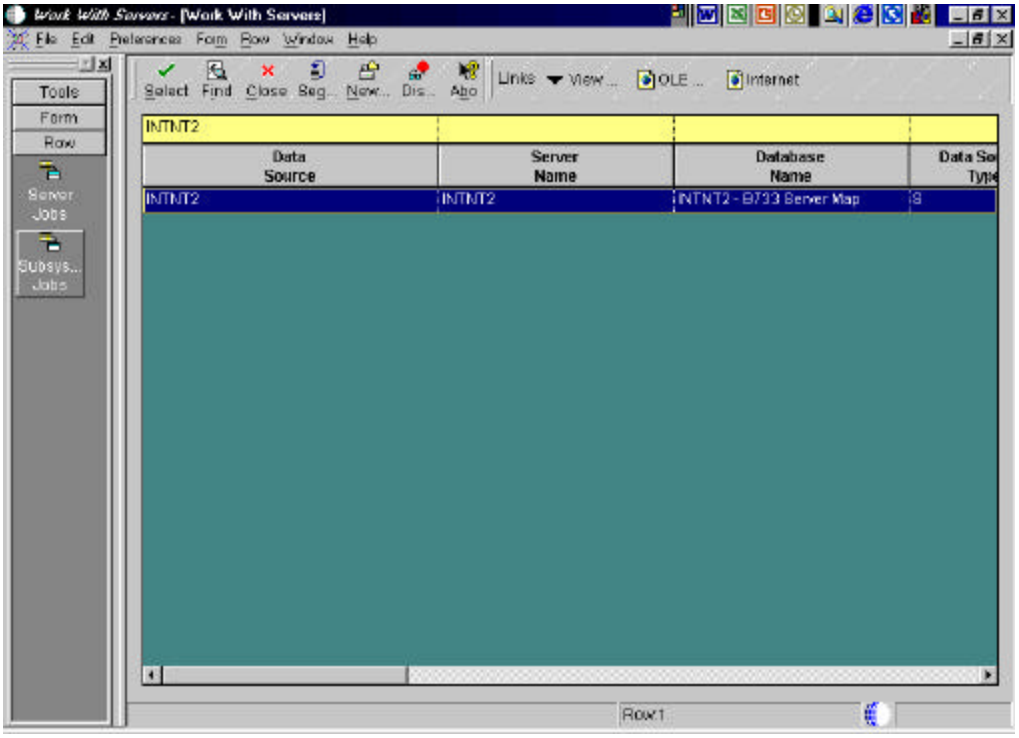
Tips and Techniques
Before you start this OneWorld subsystem on the UNIX-based Enterprise Server, you must first bring up Services. Also, you should have already updated the jde.ini file on the Enterprise Server to point to the new queue.

1. From Batch Versions, select R00460.



2. Select version XJDE0001 and then click submit.

3. You can verify the status of the subsystem job using Work With Servers.



4. Inquire on your server and select subsystem jobs.