JD Edwards EnterpriseOne Tools

Messaging Queue Adapter Configuration Guide for WebSphere MQ on UNIX Systems

9.2

JD Edwards EnterpriseOne Tools
Messaging Queue Adapter Configuration Guide for WebSphere MQ on UNIX Systems

9.2

Part Number: E64143-04

Contents

Preface i

1 Introduction 1

Introduction 1
Accessing Minimum Technical Requirements 1
Creating Outbound WebSphere MQ Queues For Real-Time Events 2
Configuring EnterpriseOne to Support WebSphere MQ Version 6.0 and Version 7.0 3

2 Setting Up Inbound and Outbound WebSphere MQ Queues for Z Events Only: Step 1 5

Setting Up Inbound and Outbound WebSphere MQ Queues for Z Events Only: Step 1 5
Configuring the JD Edwards EnterpriseOne Server Components 5
Communicating Between Systems - Starting Up Channels And Listeners 8
Working with WebSphere MQ Queue Manager Commands 10

3 Configuring the JD Edwards EnterpriseOne Server jde.ini File for Z Events Only: Step 2 11

Configuring the JD Edwards EnterpriseOne Server jde.ini File for Z Events Only: Step 2 11

4 Setting Up UBE Queues for Z Events Only: Step 3 13

Setting Up UBE Queues for Z Events Only: Step 3 13
Setting up the JD Edwards EnterpriseOne Server to Support Multiple UBE Queues 13
Setting up a Client to Support the R00460 Subsystem UBE 13

5 Configuring the JD Edwards EnterpriseOne Server: Step 4 15

Configuring the enterpriseone.sh Script on the JD Edwards EnterpriseOne Server 15

6 Configuring the Interoperability Features for Z Events Only: Step 5 17

Configuring the Interoperability Features for Z Events Only: Step 5 17

Setting Up Flat File Cross-Reference 17
Setting Up Data Export Controls 18
Setting Up the Sales Order Entry Application 19
Configuring the Address Book Revisions Application 20
Configuring the Address Book Master Business Function 21

7 Restarting the Services: Step 6 23

Restarting the Services: Step 6 23

8 Running the R00460 Subsystem for WebSphere MQ for Z Events Only: Step 7 25

Running the R00460 Subsystem for WebSphere MQ for Z Events Only: Step 7 25
Starting the R00460 Subsystem 25
Viewing Subsystems Running on a Server 26
Terminating Subsystems 27
Verifying Event Delivery 27

Preface

Welcome to the JD Edwards EnterpriseOne documentation.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at *http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc* .

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit *http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info* or visit *http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs* if you are hearing impaired.

Related Information

For additional information about JD Edwards EnterpriseOne applications, features, content, and training, visit the JD Edwards EnterpriseOne pages on the JD Edwards Resource Library located at:

http://learnjde.com

Conventions

The following text conventions are used in this document:

Convention	Meaning
Bold	Boldface type indicates graphical user interface elements associated with an action or terms defined in text or the glossary.
Italics	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`Monospace`	Monospace type indicates commands within a paragraph, URLs, code examples, text that appears on a screen, or text that you enter.
> Oracle by Example	Indicates a link to an Oracle by Example (OBE). OBEs provide hands-on, step- by-step instructions, including screen captures that guide you through a process using your own environment. Access to OBEs requires a valid Oracle account.

1 Introduction

Introduction

Oracle's JD Edwards EnterpriseOne Adapter for IBM WebSphere MQ enables you to connect third-party applications to EnterpriseOne by sending and receiving messages through IBM's WebSphere MQ messaging system. The adapter monitors an inbound queue for request and reply messages, performs the requested services, and places the results on outbound queues. The adapter also monitors JD Edwards EnterpriseOne for certain activities and publishes the results in an outbound message queue. All messages transported through IBM WebSphere MQ are in the form of XML documents.

The purpose of this document is to describe the configuration and operation of the adapter. A separate document, the *JD Edwards EnterpriseOne Tools Interoperability Guide* , in the chapter “Understanding Messaging Queue Adapters" explains the design of the adapter, the formats of the documents, and the tasks necessary to create, modify, and process the XML documents in the Events chapters.

The JD Edwards EnterpriseOne Adapter for IBM WebSphere MQ is an EnterpriseOne product that can be licensed and installed independently. You use this adapter to connect the JD Edwards EnterpriseOne system with any system that can implement the IBM WebSphere MQ messaging protocols, including IBM WebSphere Commerce Suite (WCS), and produce and consume XML documents in the prescribed formats. The JD Edwards EnterpriseOne Adapter for WebSphere MQ exports and imports XML documents through WebSphere MQ in the prescribed formats.

Overview of three operations:

1. Inbound transactions through calling business functions (BSFNs).
2. Outbound Z events delivery through setting up EnterpriseOne Z events mechanism.

The remaining guide applies only to inbound and outbound using Z events.

1. Outbound Real-Time events delivery through setting up EnterpriseOne Real-Time events mechanism.

Accessing Minimum Technical Requirements

Customers must conform to the supported platforms for the release as detailed in the JD Edwards EnterpriseOne Minimum Technical Requirements. In addition, JD Edwards EnterpriseOne may integrate, interface, or work in conjunction with other Oracle products. Refer to the following link for cross-reference material in the Program Documentation for Program prerequisites and version cross-reference documents to assure compatibility of various Oracle products: `http://www.oracle.com/corporate/contracts/index.html`

You can locate the JD Edwards EnterpriseOne Tools Release 9.2 Certifications from My Oracle Support.

To access JD Edwards EnterpriseOne Tools Release 9.2 Certifications:

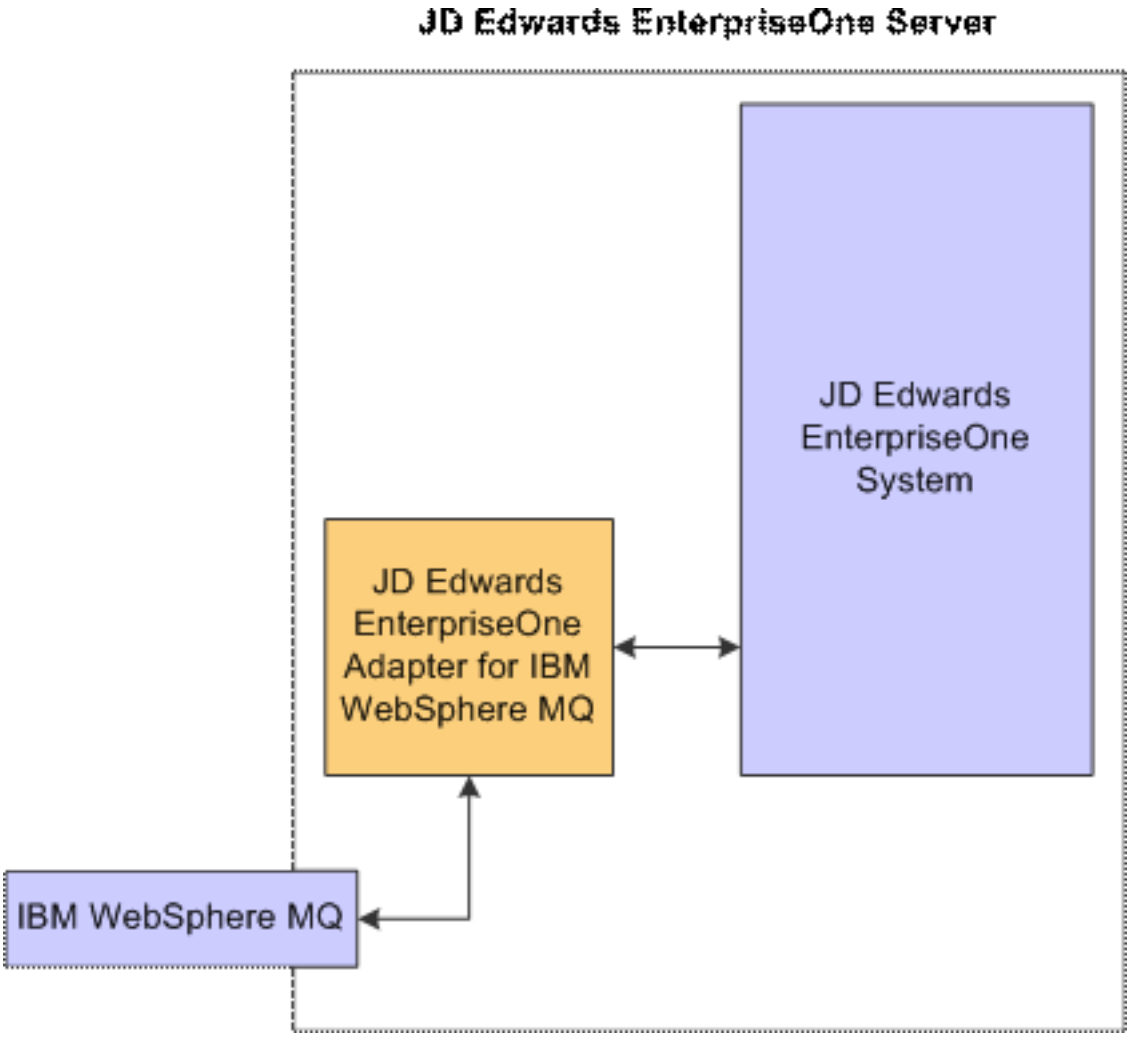
1. Navigate to My Oracle Support (https://support.oracle.com).
2. Click the Certifications tab.
3. Search for the product.

Creating Outbound WebSphere MQ Queues For Real-Time Events

If you are interested in receiving real-time events, real-time event queues must be created. The real-time event queue does not need to be created if you are not interested in receiving real-time events.

Note: "Creating a MSMQ Real-Time Event Queue" in the *JD Edwards EnterpriseOne Tools Interoperability Guide* .

This illustration shows the JD Edwards EnterpriseOne server with the IBM WebSphere MQ adapter.



Prerequisites

See *Accessing Minimum Technical Requirements*.

Configuring EnterpriseOne to Support WebSphere MQ Version 6.0 and Version 7.0

Note: These steps are not needed for WebSphere MQ 8.0. Starting with Tools Release 9.1.5.7, only one version of these files are delivered, without a version number in the name. The libraries delivered without version names are only compatible with WebSphere MQ 8.0 starting with Tools Release 9.1.5.7 and cannot be used with older versions of WebSphere MQ. Conversely, the driver libraries with names ending in *_6 or *_7 cannot be used with WebSphere MQ 8.0.

To properly perform this task, you must be logged on as the UNIX user who starts JD Edwards EnterpriseOne services (for example, jde900).

1. Use this command to change your directory:

   cd $SYSTEM/lib
2. When using the 6.0 version of WebSphere MQ, create a soft link to the WebSphere MQ 6.0 version of the driver libraries. These file names end in *_6.so (use .sl instead of .so on HP-UX PA-RISC servers).

   ln -s libmqnotify_6.so libmqnotify.so

   ln -s libmqsadapt_6.so libmqsadapt.so

   ln -s libmqsrtdrv_6.so libmqsrtdrv.so
3. When using the 7.0 version of WebSphere MQ, create a soft link to the WebSphere MQ 7.0 version of the driver libraries. These file names end in *_7.so (use .sl instead of .so on HP-UX PA-RISC servers).

   ln -s libmqnotify_7.so libmqnotify.so

   ln -s libmqsadapt_7.so libmqsadapt.so

   ln -s libmqsrtdrv_7.so libmqsrtdrv.so

2 Setting Up Inbound and Outbound WebSphere MQ Queues for Z Events Only: Step 1

Setting Up Inbound and Outbound WebSphere MQ Queues for Z Events Only: Step 1

IBM WebSphere MQ is a queue messaging system that requires a sender and receiver relationship. One side of IBM WebSphere MQ is installed on Oracle's JD Edwards EnterpriseOne server while the related side is installed on another physical or logical machine. The setup on the JD Edwards EnterpriseOne server enables EnterpriseOne to receive inbound messages from a third-party application or system and to place outbound messages in a queue for processing by the third-party application or system. Refer to the applicable IBM documentation for instructions on installing WebSphere MQ on other machines.

The procedures described in this section assume you have already installed the IBM WebSphere MQ product onto your JD Edwards EnterpriseOne server.

You must set up your JD Edwards EnterpriseOne server so that the system can receive inbound messages from a third-party application or system and then place responses, in the form of outbound messages, in a queue for processing by the third-party application or system.

In addition to information about setting up your JD Edwards EnterpriseOne server, this section also provides examples for:

- Communicating between the JD Edwards EnterpriseOne IBM WebSphere MQ Server and the Non-JD Edwards EnterpriseOne IBM WebSphere MQ Server.
- Working with IBM WebSphere MQ Queue Manager Commands.

Note: On all systems, you must add each user that will connect to the WebSphere MQ instance (for example jde910) to the mqm group in /etc/group.

Configuring the JD Edwards EnterpriseOne Server Components

Perform the following tasks to set up your JD Edwards EnterpriseOne server so that EnterpriseOne can receive an inbound message from another system or application and respond by placing an outbound message in the outbound queue. If your JD Edwards EnterpriseOne server is an IBM pSeries machine or an HP-UX machine, use the appropriate setup task to set up your machine then proceed to the task titled *To create and start the IBM WebSphere MQ Queue*

Manager. If you have other UNIX platforms, go directly to the task titled *To create and start the IBM WebSphere MQ Queue Manager.*

- Prepare for IBM WebSphere MQ version 6.0 or version 7.0.
- Set up your pSeries machine.
- Set up your HP machine.
- Create and start IBM WebSphere MQ queue manager.
- Create MQ local queues.
- Create a MQ remote queue.
- Create a MQ local sender channel.
- Create a MQ local receiver channel.
- End the IBM WebSphere MQ queue manager.

CAUTION: The names of queues and channels in IBM WebSphere MQ are case-sensitive. Be sure to use capital letters 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 exactly match the queue names you specify in the jde.ini file on the JD Edwards EnterpriseOne server.

To setup an IBM AIX (pSeries) machine for WebSphere MQ 6 or MQ 7

Note: Perform this setup task only if your JD Edwards EnterpriseOne server is an IBM AIX (pSeries) machine when using WebSphere MQ 6 or MQ 7. These steps are not required with WebSphere MQ 8.

To properly perform this task, you must be logged on with root authority.

1. Edit the /etc/services file and add this line:
   ibm-mqseries 1414/tcp # WebSphere MQ channel listener
2. Edit the /etc/inetd.conf file and add this line:
   ibm-mqseries stream tcp nowait mqm /usr/mqm/bin/amqcrsta amqcrsta -m JDE_QMGR
3. After you have made the file edits, you must either refresh the *inetd* process or kill the process and restart it. Do one of these:
   - To refresh the inetd process, enter this command:
     refresh -s inetd
   - To kill the inetd process, you must first locate it using this command:
     ps -ef | grep inetd
     After you have located the inetd process, you can kill it using this command:
     kill -1 inetd *<process_ID>*
     where *<process_ID>* is the ID you determined using the grep command.
4. Edit the .profile files for each user that will connect to the WebSphere MQ instance (may include jde900, mqm, and/or root) to add this export line:
   MQSERVER=SYSTEM.DEF.SVRCONN/TCP/<hostname>
5. Log out these users from step 4, and log them back in to execute the .profile to set MQSERVER.

To setup your HP machine

> **Note:** Perform this setup task only if your JD Edwards EnterpriseOne server is an HP machine.

To properly perform this task, you must be logged on as the UNIX user who starts JD Edwards EnterpriseOne services (for example, jde910).

1. Edit the .profile file to add the following export line:
   export MQS_NO_SYNC_SIGNAL_HANDLING=
2. Log out and back in to execute the .profile to set MQS_NO_SYNC_SIGNAL_HANDLING.

To create and start WebSphere MQ Queue Manager

1. On your JD Edwards EnterpriseOne server, open a shell.
2. To create the queue manager, execute this command:
   crtmqm -q JDE_QMGR
   This establishes a JDE queue manager as the default queue manager.
3. To start the queue manager, execute this command:
   strmqm JDE_QMGR

To create MQ local queues

You must create these local queues:

- INBOUND.Q
- SUCCESS.Q
- ERROR.Q
- DEFRES.Q
- OUTBOUND.Q.XMIT

1. Access the shell you used to start the queue manager.
2. To start the MQSC facility, enter this command:
   runmqsc JDE_QMGR
3. To create the local queues, run these IBM WebSphere MQ commands:
   Define QLOCAL(INBOUND.Q) DEFPSIST(YES)
   DESCR('Queue for messages into EnterpriseOne')
   Define QLOCAL(SUCCESS.Q) DEFPSIST(YES)
   DESCR('Queue for successful messages in EnterpriseOne')
   Define QLOCAL(ERROR.Q) DEFPSIST(YES)
   DESCR('Queue for error messages in EnterpriseOne')
   Define QLOCAL(DEFRES.Q) DEFPSIST(YES)
   DESCR('Queue for responses if not in message in EnterpriseOne')
   Define QLOCAL(OUTBOUND.Q.XMIT) DEFPSIST(YES)

USAGE(XMITQ) DESCR('Transmit queue to NC system')

To create the MQ remote queue

In the same shell you used in the previous task, enter this 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('EnterpriseOne outbound queue to NC system')

> **Note:** ECE_IN2MQI is the third-party remote queue name and ECE_MQI_QMGR is the third-party queue manager name.

To create a MQ local sender channel

In the same shell you used in the previous task, enter this 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

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 third-party application or system')

To end the IBM WebSphere MQ command mode

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

END

This exits the runmqsc program.

Communicating Between Systems - Starting Up Channels And Listeners

You must set up communications between the JD Edwards EnterpriseOne IBM WebSphere MQ Server (functioning on the EnterpriseOne server) and the third-party IBM WebSphere MQ server. This section provides example tasks for setting up communications. The example communications setup uses these sequenced and machine dependent steps:

- Start the Queue Manager on the third-party IBM WebSphere MQ Server.
- Start the Queue Manager on the JD Edwards EnterpriseOne IBM WebSphere MQ Server.
- Start the channel on the third-party IBM WebSphere MQ Server.
- Start the channel on the JD Edwards EnterpriseOne IBM WebSphere MQ Server.

> **Note:** The first time you start your communications channels, you might need to manually start the sender channel on the local machine and then manually start the receiver channel on the remote machine. After the initial start of the channels, when you start the sender on the local machine, the receiver on the remote machine should automatically start.

To start the Queue Manager on the third-party IBM WebSphere MQ Server

Access the third-party IBM WebSphere MQ server.

1. In a shell, enter this command to start the queue manager:

   strmqm ECE_MQI_QMGR

   > **Note:** ECE_MQI_QMGR is the third-party queue manager name.

2. In the same shell, enter these commands to start the listener:

   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 JD Edwards EnterpriseOne IBM WebSphere MQ Server

Access the JD Edwards EnterpriseOne IBM WebSphere MQ Server.

1. In a shell, enter this command to start the Queue Manager:

   strmqm ECE_MQI_QMGR

   > **Note:** ECE_MQI_QMGR is the third-party queue manager name.

2. In the same shell, enter these commands to start the listener:

   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 third-party IBM WebSphere MQ Server

Access the third-party IBM WebSphere MQ server.

In the same shell you used to start the Queue Manager, enter these commands:

runmqchl MQI2OW_CHL

runmqchl OW2MQI_CHL

To start the channel on the JD Edwards EnterpriseOne IBM WebSphere MQ server

Access the JD Edwards EnterpriseOne IBM WebSphere MQ server.

In the same shell you used to start the Queue Manager, enter these commands:

runmqchl OW2MQI_CHL

runmqchl MQI2OW_CHL

Working with WebSphere MQ Queue Manager Commands

This table provides example commands that you can use when working with IBM WebSphere MQ Queue Manager commands.

> **CAUTION:** The names of queues and channels in IBM WebSphere MQ are case-sensitive. Be sure to use capital letters 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 JD Edwards EnterpriseOne server.

Command	Description
strmqm JDE_QMGR	Start the queue manager.
endmqm -i JDE_QMGR	End the queue manager. The -i switch means immediately.
runmqsc JDE_QMGR	Start using IBM WebSphere MQ commands.
display QL(*qname*.Q)	Display the local queue.
clear QL(*qname*.Q)	Clear the local queue.
END	Stop using IBM WebSphere MQ commands
amqsbcg *qname*.Q JDE_QMGR	Run the browse queue command. This does not remove the message.
runmqchl(channel_name)	Start the channel.
runmqslr <*Gateway Name*> TpName JDE_QMGR	Start a listener. <*Gateway Name*> is the machine on which the listener is being run.
resetchl (channel_name)	Reset a channel that has become inactive.

3 Configuring the JD Edwards EnterpriseOne Server jde.ini File for Z Events Only: Step 2

Configuring the JD Edwards EnterpriseOne Server jde.ini File for Z Events Only: Step 2

Note: Make sure the JD Edwards EnterpriseOne services are stopped until after you make the requisite changes to the jde.ini file and the enterpriseone.sh script on Oracle's JD Edwards EnterpriseOne server.

The JD Edwards EnterpriseOne server jde.ini file must be configured as follows:

Parameter	Description
[SECURITY]	You must have these values set in order for JD Edwards EnterpriseOne to accept messages.
User=JDE	
Password=JDE	
[JDENET_KERNEL_DEF17]	You must update the DEF value to next number. For JD Edwards EnterpriseOne, this value is 17.
krnlName=MQSI Kernel	
dispatchDLLName=libmqsadapt.so	Use libmqsadapt.sl instead of libmqsadapt.so on HPUX PA-RISC
dispatchDLLFunction=JDEK_ DispatchMQSeries Process	
maxNumberOfProcesses=1	

Parameter	Description
[MQSI]	These settings are for the header information on the message that is required for IBM WebSphere Commerce Integrator.
QMGRName=JDE_QMGR QInboundName=INBOUND.Q QErrorName=DEFRES.Q	The name of the queues can be any name, but must match the names you specify in the WebSphere MQ queue setup.

Parameter	Description
QOutboundName=OUTBOUND.Q	
TimeoutWaitInterval=15	
MaxBufferLength=100,000	The value of this field is measured in bytes. The maximum value that you can set is 100,000,000. The default value is 100,000. You should not use the maximum value, as this can cause performance issues. You can use the IBM WebSphere MQ tools to determine the average size of a message and then set the MaxBufferLength value based on your needs.
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	If you use the adapter without IBM WebSphere Commerce Integrator, you can specify the create header as equal to No, and you should set the IBM WebSphere 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.
OWHostName=*host_name*	You must specify a host name. The name you specify is the machine used to create the net message to trigger the outbound adapter. This is the server on which JD Edwards EnterpriseOne resides.

4 Setting Up UBE Queues for Z Events Only: Step 3

Setting Up UBE Queues for Z Events Only: Step 3

You must set up at least two UBE queues. This is required because Oracle's JD Edwards EnterpriseOne Adapter for WebSphere MQ 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 JD Edwards EnterpriseOne Adapter for IBM WebSphere MQ subsystem.

This section describes:

- Setting up the JD Edwards EnterpriseOne server to support multiple UBE queries.
- Setting up a client to support the R00460 Subsystem UBE.

Setting up the JD Edwards EnterpriseOne Server to Support Multiple UBE Queues

Configure the JD Edwards EnterpriseOne server to support additional UBE queues through the client to run the R00460 subsystem UBE.

Setting up a Client to Support the R00460 Subsystem UBE

If you submit or start the R00460 subsystem UBE from a JD Edwards EnterpriseOne Microsoft Windows client, you must temporarily modify that client's jde.ini file. The temporary modification is required so that the client can direct the R00460 subsystem to the appropriate UBE queue name. After the client submits or starts the R00460 subsystem UBE, you must undo the temporary change so that the 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 setting is correct:

Parameter	Description
[NETWORK QUEUE SETTINGS]	Defines the name of the local or server-based UBE queue.

Parameter	Description
UBEQueue=	If you want the client to start or submit the R00460 subsystem, enter a value that corresponds with the value set by the UBEQueue1= on the JD Edwards EnterpriseOne server. If you want the Microsoft Windows client to use the normal server-based UBE processing queue, enter a value that corresponds with the equivalent value for that queue on the JD Edwards EnterpriseOne server.

CAUTION: Any time you modify settings in the jde.ini file on the Microsoft Windows client, you must exit and restart JD Edwards EnterpriseOne in order for those changes to become effective.

5 Configuring the JD Edwards EnterpriseOne Server: Step 4

Configuring the enterpriseone.sh Script on the JD Edwards EnterpriseOne Server

You must check the UNIX script file called enterpriseone.sh that is located in the $SYSTEM/SharedScripts directory. This script is generated by the initial Oracle's JD Edwards EnterpriseOne server installation program. It is used by the UNIX system to read and set the JD Edwards EnterpriseOne environment variables. You must edit the script to add an environment variable for the JD Edwards EnterpriseOne Adapter for WebSphere MQ if it is not already present.

To configure the enterpriseone.sh script

Access the JD Edwards EnterpriseOne server.

1. Shut down JD Edwards EnterpriseOne before you edit the enterpriseone.sh script file.
2. Locate the enterpriseone.sh script file in the $SYSTEM/SharedScripts directory.
3. To open the enterpriseone.sh script file, run the following command:

   vi $SYSTEM/SharedScripts/enterpriseone.sh
4. Add the following line commands to the enterpriseone.sh script if they are not already present:

   ICU_DATA=$SYSTEM/locale/xml/

   export ICU_DATA
5. Save and close the enterpriseone.sh script file.
6. Log out of your current user session.
7. Log back onto your user session to enable the JD Edwards EnterpriseOne Adapter for IBM WebSphere MQ environment variable. Restart the Server Manager Agent using the restartAgent script.

6 Configuring the Interoperability Features for Z Events Only: Step 5

Configuring the Interoperability Features for Z Events Only: Step 5

You can use Z, real-time, or XAPI events to receive transactions from Oracle's JD Edwards EnterpriseOne. To use real-time events, you must set up your system as indicated in the *JD Edwards EnterpriseOne Tools Interoperability Guide*, and you must set up events as indicated in the *JD Edwards EnterpriseOne Applications Business Interface Reference Guide*.

Z event outbound processing uses interface table (Z-table) processes supported by JD Edwards EnterpriseOne applications. The outbound processes available to you vary depending on which JD Edwards EnterpriseOne release you use. If you use interface tables for transaction-specific outbound processing, you must perform configuration tasks such as setting up processing options and setting up data export controls.

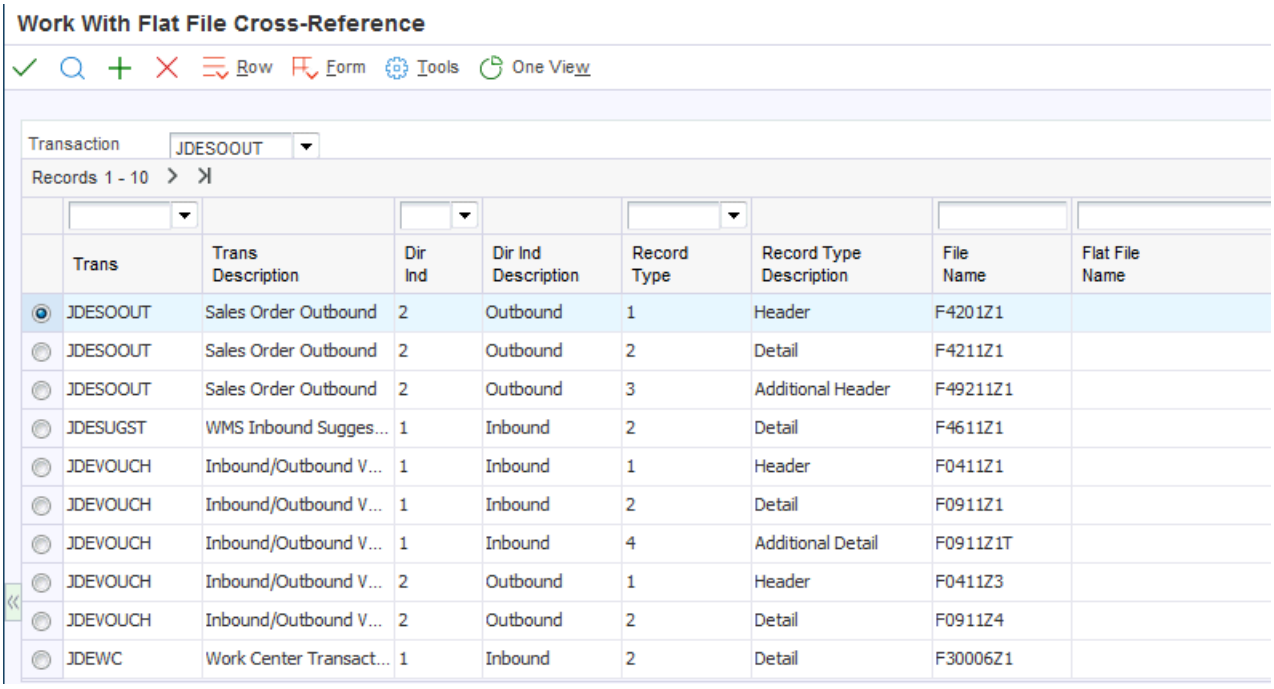
This section provides guidance for setting up the data export controls and also provides examples for setting up JD Edwards EnterpriseOne applications that support Z event processing. The application examples include the following:

- Setting Up Flat File Cross-Reference (P47002)
- Setting Up Data Export Controls
- Setting up the Processing Options for the Sales Order Master Business Function (P4210).
- Setting up the Processing Options for Address Book Revisions (P01012).
- Setting up the Processing Options for the Address Book Master Business Function (P0100041).

Setting Up Flat File Cross-Reference

When you enable Z events, you also update the Flat File Cross-Reference (F47002) table. The transaction type that you entered in the processing option maps to table F47002 to determine which interface tables to use to retrieve the information. You use the Flat File Cross-Reference program (P47002) to update table F47002.

Setting up flat file cross-reference



Trans	Trans Description	Dir Ind	Dir Ind Description	Record Type	Record Type Description	File Name	Flat File Name
JDESOOUT	Sales Order Outbound	2	Outbound	1	Header	F4201Z1	
JDESOOUT	Sales Order Outbound	2	Outbound	2	Detail	F4211Z1	
JDESOOUT	Sales Order Outbound	2	Outbound	3	Additional Header	F49211Z1	
JDESUGST	WMS Inbound Sugges...	1	Inbound	2	Detail	F4611Z1	
JDEVOUCH	Inbound/Outbound V...	1	Inbound	1	Header	F0411Z1	
JDEVOUCH	Inbound/Outbound V...	1	Inbound	2	Detail	F0911Z1	
JDEVOUCH	Inbound/Outbound V...	1	Inbound	4	Additional Detail	F0911Z1T	
JDEVOUCH	Inbound/Outbound V...	2	Outbound	1	Header	F0411Z3	
JDEVOUCH	Inbound/Outbound V...	2	Outbound	2	Detail	F0911Z4	
JDEWC	Work Center Transact...	1	Inbound	2	Detail	F30006Z1	

Refer to the *JD Edwards EnterpriseOne Applications Data Interface for Electornic Data Interchange Implementation Guide* for instructions on setting up the flat file cross-reference application.

Setting Up Data Export Controls

You must create a Data Export Control (F0047) record for each transaction type. The record specifies the vendor-specific UBE or function to call to process the transaction. For example, for transaction type JDESOOUT you must set up a record for each order type that you want to export.

Setting Up Data Export Controls

From the Navigator, select EnterpriseOne Menus > Order Management > Sales Order Management > Sales Order Advanced & Technical Ops > Sales Interoperability > Data Export Controls.

An alternative way to access the Data Export Controls program is to type P0047 on the Fast Path.

Data Export Controls form

This table shows the values for the Data Export Controls form fields:

Form Element	Description
Trans	The values you specify in this field must match the values you have configured for JD Edwards EnterpriseOne outbound processing in the various JD Edwards EnterpriseOne applications. These are example transaction types: JDEAB JDEIL

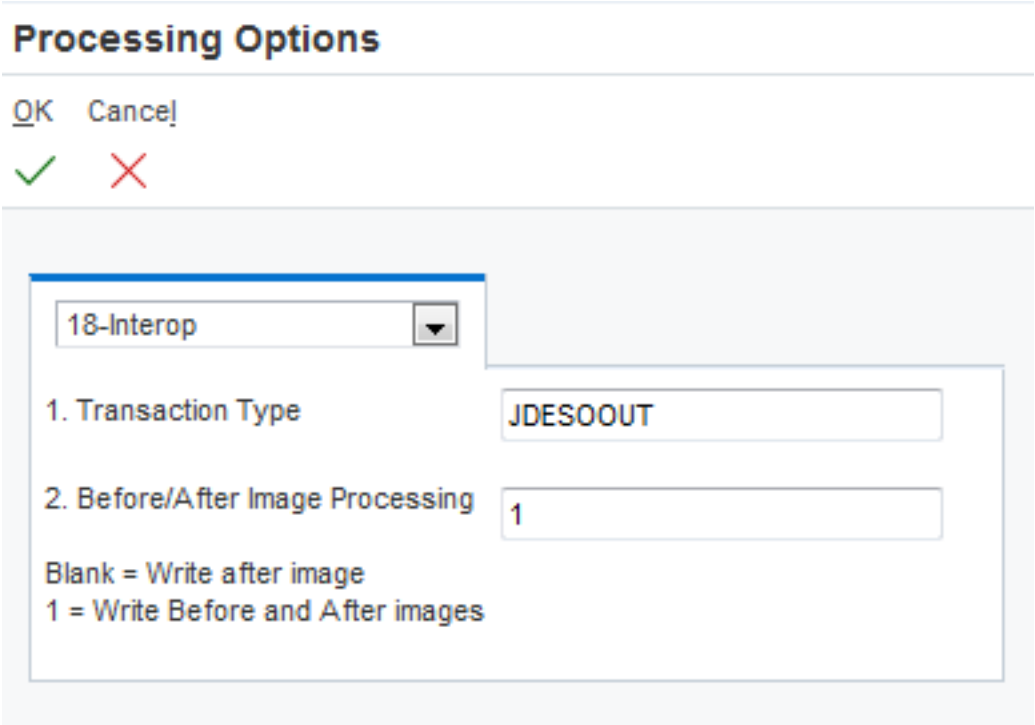
Form Element	Description
	JDEPRICE JDESOOUT
Or Ty	The values you specify in this field must match the values that are required for your system. For example, for WCS these order types are required: JDESOOUT S4 JDESOOUT SO
Seq	1.00
UBE Name	Leave this field blank.
Version	Leave this field blank.
Function Name	MQOutboundNotify.
Function Library	/u01/jdedwards/e910/system/lib/libmqnotify.so This location is an example. The actual location varies by individual site.
Execute for Add	1
Execute for Upd	1
Execute for Del	1
Execute for Inq	1
Flat File Exp Mode	0
Ext DB Exp Mode	0
Ext API Exp Mode	0
Launch Immediately	1

Setting Up the Sales Order Entry Application

You setup the JD Edwards EnterpriseOne Sales Order Entry (P4210) program to enable it for interoperability operations.

To set up the processing options for the Sales Order Entry Master Business Function (P4210)

From the Navigator, select EnterpriseOne Menus > Order Management > Sales Order Management > Daily Sales Order Management Processing > Sales Order Processing > right-click on Sales Order Detail and then select Values. On Processing Options, select 18-Interop from the 1-Defaults drop-down list.



Sales Order Detail Processing Options form

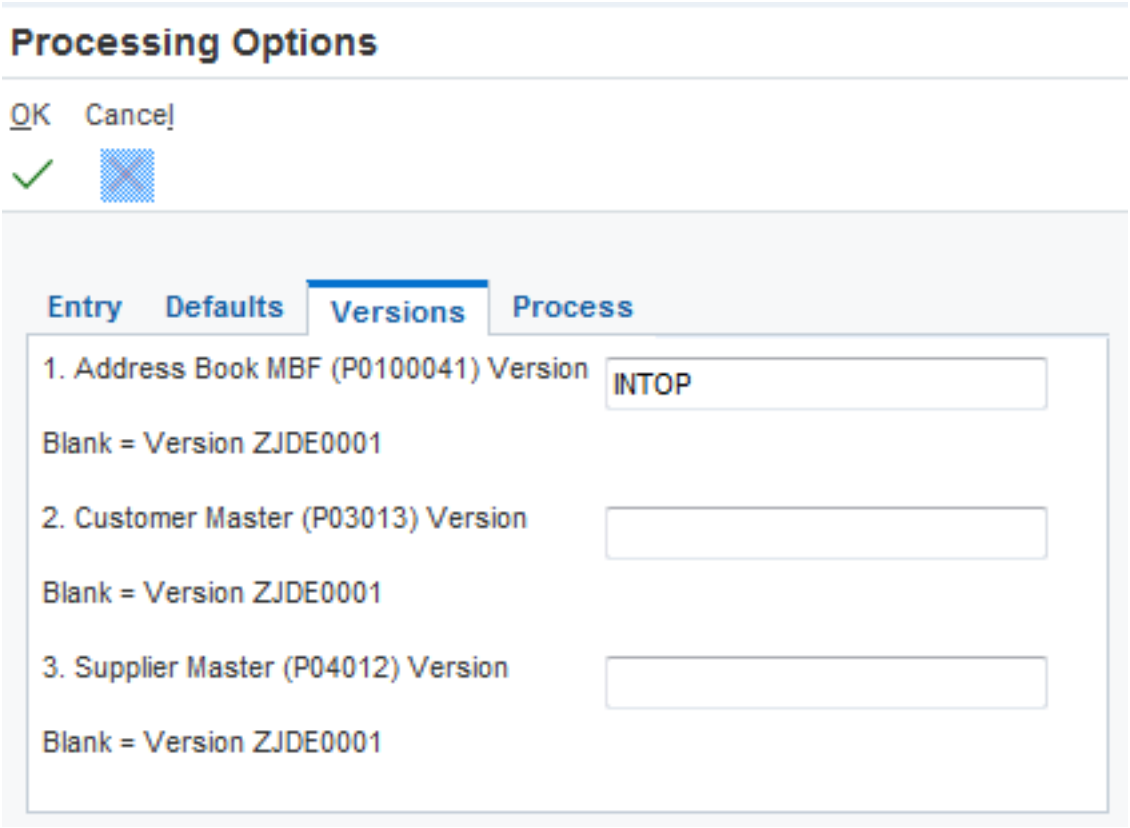
1. To define the transaction type, type JDESOOUT in the Transaction Type field.
2. To define before or after image processing, enter the appropriate value in the Before/After Image Processing field.
3. Click OK.

Configuring the Address Book Revisions Application

You set up the JD Edwards EnterpriseOne Address Book Revision (P01012) program to enable it for interoperability operations.

To set up the processing options for Address Book Revisions

From the Navigator, select EnterpriseOne Menus > Foundation Systems > Address Book > Daily Processing > right-click on Address Book Revisions and then select Values. On Processing Options, select the Versions tab.



Address Book Revisions Processing Options form

1. On Processing Options, type INTOP in the Address Book MBF (P0100041) Version field.
2. Click OK.
3. On Work With Addresses, click Close.

Configuring the Address Book Master Business Function

You set up the Address Book Master Business Function (P0100041) to enable interoperability operations.

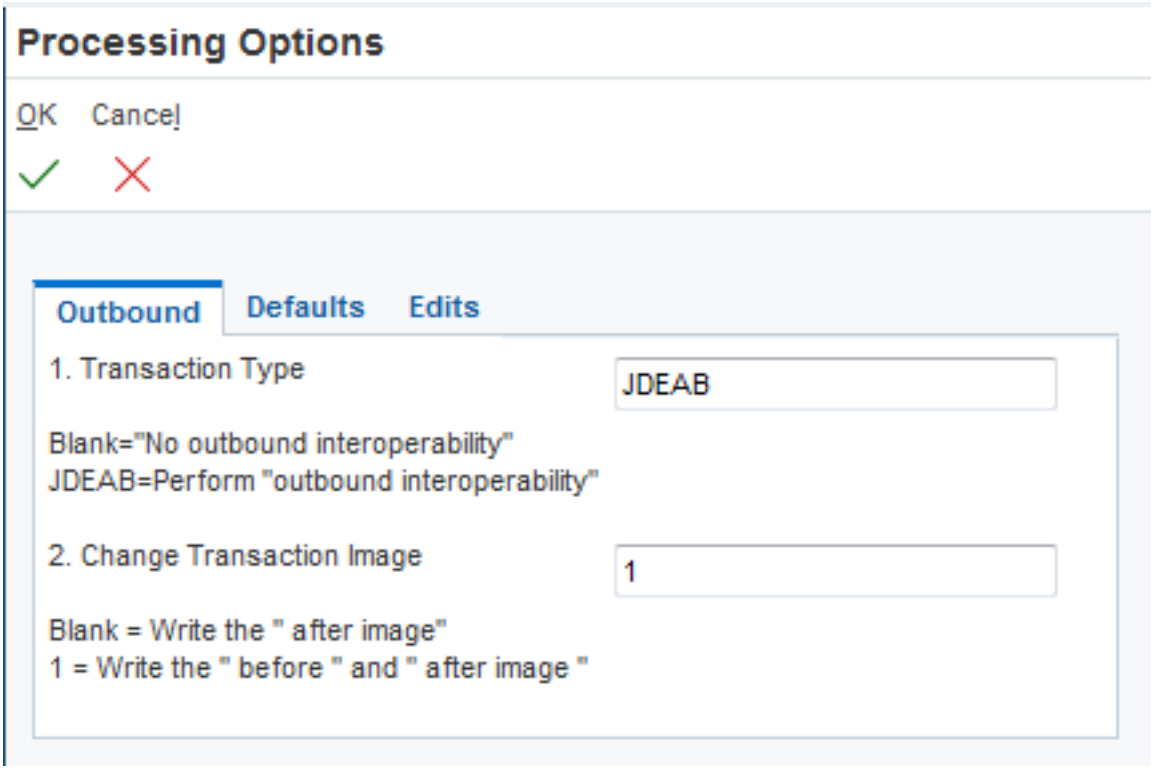
To set up the processing options for the Address Book Master Business Function

Type IV in the Fast Path.

1. On Interactive Versions -- Work With Versions, type P0100041 in the Application field, and then click Find.

2. Select the INTOP version, and then select Processing Options from the Row menu.

   If there is not a version called INTOP, select version ZJDE0001 and copy it. Call the new version INTOP and click OK.



   Address Book Master Business Function Processing Option form

3. On Processing Options, select the Outbound tab.
4. Type JDEAB in the Transaction Type field.
5. To define before or after image processing, enter the appropriate value in the Before/After Image Processing field.
6. Click OK.
7. Click Close.

7 Restarting the Services: Step 6

Restarting the Services: Step 6

After you complete the IBM WebSphere MQ queue setup and the associated configuration tasks, you can restart the UNIX services.

First you restart Oracle's JD Edwards EnterpriseOne Services for the JDE E910 service. This can be done through Server Manager or from the command line by entering the following commands:

```
cd $SYSTEM/bin32 ./RunOneWorld.sh
```

After your restart the UNIX services, you should successfully complete a PORTTEST.

8 Running the R00460 Subsystem for WebSphere MQ for Z Events Only: Step 7

Running the R00460 Subsystem for WebSphere MQ for Z Events Only: Step 7

The R00460 subsystem monitors the MSMQ queues for messages and manages Oracle's JD Edwards EnterpriseOne side of the queues. After the R00460 subsystem is started, you can verify that the subsystem is running. You can also review the job record for the subsystem. After the records are processed, you must manually terminate the subsystem job.

CAUTION: After the records are processed, instead of ending the job, subsystem jobs look for new data in the data queue. Subsystem jobs run until you terminate them.

Note:

- *"Managing JD Edwards EnterpriseOne Subsystems" in the JD Edwards EnterpriseOne Administration Guide* .

Starting the R00460 Subsystem

The R00460 subsystem must be manually started. Usually the system administrator or manager-level user is responsible for this task.

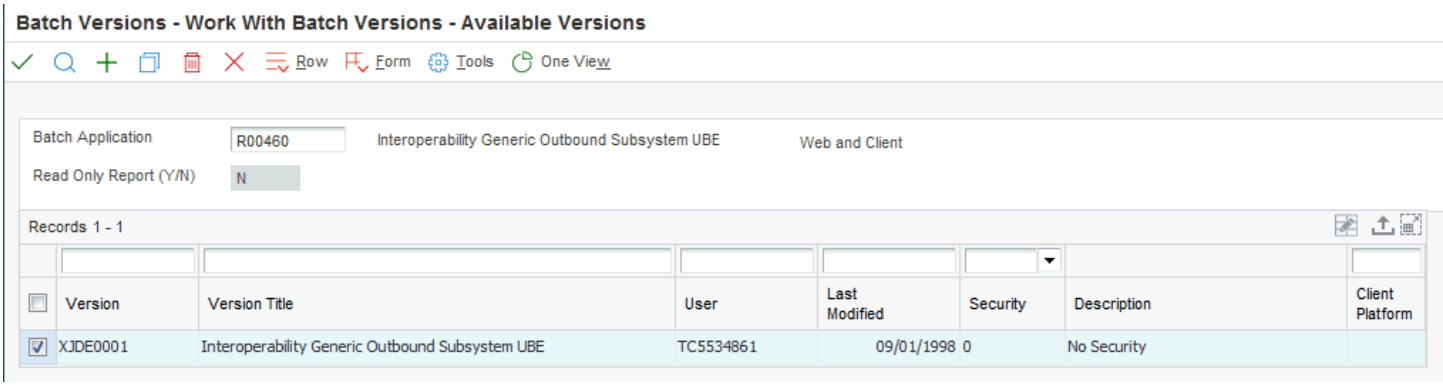
Note: Before you start the R00460 subsystem, you should have already updated the jde.ini file on the JD Edwards EnterpriseOne server to point to the new queue.

To start the R00460 subsystem

From the navigator, select EnterpriseOne Menus > EnterpriseOne Life Cycle Tools > Report Management > Batch Versions.

An alternate way to access the Batch Versions form is to type BV in the Fast Path.

1. On Work With Batch Versions, type R00460 in the Batch Application field, and then click Find.



2. Select version XJDE0001, and then click Select.
3. On Version Prompting, click Submit.
4. On Report Output Destination, select the destination option for your report, and then click OK.

Viewing Subsystems Running on a Server

You use Work With Servers to determine which subsystems are currently running or waiting on a particular server. The running subsystems are identified by report number and version.

To view subsystems running or waiting on a server

From the JD Edwards EnterpriseOne Navigator, select EnterpriseOne Menus > EnterpriseOne Life Cycle Tools > System Administration Tools > Data Source Management > Work With Servers. On the Submitted Job Search form, select Advanced from the Form menu.

1. On Work With Servers, click Find to locate all servers or use the query by example row to locate a specific server.
2. Select the server with which you want to work.
3. From the Row menu, select Subsystem Jobs.
4. On Work With Server Jobs, click one of these options:
    - Processes

      A process is a subsystem that is waiting for work. This is identified by an S (subsystem job) value in the Job Type field.
    - Waiting Jobs

      Waiting jobs are report jobs that are queued for a subsystem. This is identified by an R (subsystem record) value in the Job Type field.

All currently running JD Edwards EnterpriseOne subsystems are displayed. The status of each subsystem is shown by codes in these fields:

- Job Type

  This field indicates whether the status is a subsystem record or a subsystem job. Valid values are:

  - R or subsystem record
  - S or subsystem job

- Job Status
  - W - subsystem record waiting
  - P - subsystem record processing
  - E - subsystem record to end the job
  - R - subsystem job running

Terminating Subsystems

You must manually terminate subsystem jobs. Two methods of termination are available:

- Stopping a subsystem job causes it to terminate after it finishes processing the current record. Additional unprocessed records in the F986113 table will not be processed, and no new records can be written. Essentially the unprocessed records will be lost; that is, the process that initiated the record is not notified that the record was not processed.
- Ending a subsystem job causes it to terminate after processing all of the existing subsystem records. No new records can be written to the F986113 table.

To stop subsystems

Either Fast Path to "WSJ" or on the JD Edwards EnterpriseOne Menu, select JD Edwards EnterpriseOne Life Cycle Tools > System Administration Tools > System Administration Tools > Data Source Management > Work With Servers.

1. On Work With Servers, click Find.
2. Select the server in the detail area, and then select Subsystem Jobs from the Row menu.
3. On Work With Subsystems, locate a running subsystem.
4. Select the running subsystem that you want to stop, and then select Stop Subsystem from the Row menu.
5. On End Subsystem Job, click OK.

To end subsystems

Either Fast Path to "WSJ" or on the JD Edwards EnterpriseOne Menu, select JD Edwards EnterpriseOne Life Cycle Tools > System Administration Tools > System Administration Tools > Data Source Management > Work With Servers.

1. On Work With Servers, click Find.
2. Select the server in the detail area, and then select Subsystem Jobs from the Row menu.
3. On Work With Subsystems, locate a running subsystem.
4. Select the running subsystem that you want to end, and then select End Subsystem Job from the Row menu.
5. On End Subsystem Job, click OK.

Verifying Event Delivery

Once finished with the installation, you should verify the event delivery. Use the WebSphere MQ Explorer to select the queue that you created to received JD Edwards EnterpriseOne events. Right-click on the queue and select Browse Messages to display the messages.

Note: “Verifying Event Delivery" in the *JD Edwards EnterpriseOne Tools Interoperability Guide*