

JD Edwards EnterpriseOne

Transaction Server Reference Guide

Release 9.1 for Microsoft Windows using Oracle WebLogic

E24429-06

December 2013

Copyright © 2011, 2013, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

Preface	vii
Documentation Accessibility	vii
Related Documents	vii
Conventions	vii
 1 Understanding the Transaction Server Components	
1.1 Overview	1-1
1.2 Certifications (Formerly Known as Minimum Technical Requirements)	1-2
1.3 Obtaining the Transaction Server Managed Software Component	1-2
1.4 Prerequisites	1-2
1.5 Understanding Transaction Server Communications	1-2
 2 Configuring the Transaction Server Components	
2.1 JDBj Database Configuration	2-1
2.1.1 JDBj Bootstrap Datasource	2-1
2.1.2 JDBj Bootstrap Session	2-2
2.1.3 JDBj Connection Pools	2-3
2.1.4 JDBC Drivers	2-4
2.1.5 JDBj Logging	2-5
2.1.6 Oracle Database Settings	2-5
2.1.7 JDBj Runtime Properties	2-6
2.1.8 JDBj Spec Datasource	2-6
2.2 Real Time Events	2-7
2.2.1 Event Processor Configuration Settings	2-7
2.2.2 Trigger Listener Kernel	2-9
2.3 Network Settings	2-9
2.3.1 JDENET Configuration	2-9
2.3.2 Security Server Configuration	2-9
2.4 Log File Configuration (jdelog.properties)	2-10
 3 Configuring JD Edwards EnterpriseOne	
3.1 Verifying User Defined Codes (UDCs)	3-1
3.2 Verifying Tables	3-1
3.3 Verifying Serialized Objects	3-2
3.4 Activating Events	3-2

3.5	Creating Triggers	3-5
4	Adding a Subscriber	
4.1	Setting Processing Options for Event Subscribers Application (P90702A)	4-1
4.2	Using Event Subscribers Application to Add a Subscriber	4-3
4.3	Working with Subscribers in JD Edwards EnterpriseOne.....	4-4
4.3.1	Adding a JAVACONN Subscriber.....	4-4
4.3.2	Creating a JMSQUEUE Subscriber.....	4-5
4.3.3	Creating a JMSTOPIC Subscriber.....	4-7
4.3.4	Creating a Subscription.....	4-9
4.3.5	Adding an Event to the Subscription.....	4-10
4.3.6	Adding Environment(s) to a Subscription.....	4-12
5	Troubleshooting Real Time Events Processing	
6	Upgrading the Transaction Server RTE Software Component Using Server Manager	
7	Uninstalling the Transaction Server RTE Software Component using Server Manager	
A	Architecture of the Guaranteed Event System	
B	Creating and Configuring Multiple Transaction Servers per Environment or User/Role	
B.1	Creating and Configuring Multiple Transaction Servers per Environment	B-1
B.2	Creating and Configuring Multiple Transaction Servers per Role/User.....	B-3
C	Working with JMS Topic or JMS Queue	
C.1	Use Case for JMS Topic	C-1
C.2	Use Case for JMS Queue	C-1
D	Testing and Validating the JMSQUEUE Configuration	
E	Creating a JMS Topic in the JD Edwards EnterpriseOne Transaction Server Application	
F	Creating a Subscriber JMS Queue in WebLogic Server other than Transaction Server	
F.1	Creating the JMS Server	F-1
F.2	Creating the JMS Module	F-3
F.3	Creating the JMS Resource	F-4

G Creating an Oracle JD Edwards EnterpriseOne Service Request for Transaction Server Support

H Clustering for the JD Edwards Transaction Server on WebLogic Server

H.1	Understanding WebLogic Clustering	H-1
H.2	Creating a Cluster on WebLogic Server	H-2
H.3	Creating the Servers.....	H-2
H.4	Creating Machines	H-5
H.5	Associate Machines with Servers	H-7
H.6	Cluster the Servers	H-10
H.7	Setup the Cluster for Load Balancing	H-12
H.8	Setup the Cluster for Failover	H-13
H.9	Installing the Transaction Server RTE Component on WebLogic Server using Server Manager H-14	
H.10	Using JDBC Persistent Stores	H-20
H.11	Create JDBC Datasources.....	H-20
H.12	Create JDBC Persistent Stores	H-26
H.13	Associate a JMS Server with a Persistent Store.....	H-29
H.14	Additional Configuration for Using Horizontal Cluster	H-31

Preface

Welcome to the JD Edwards EnterpriseOne Transaction Server Reference Guide for Microsoft Windows Oracle WebLogic.

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 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 Documents

You can access related documents from the JD Edwards EnterpriseOne Release Documentation Overview pages on My Oracle Support. Access the main documentation overview page by searching for the document ID, which is 876932.1, or by using this link:

<https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=876932.1>

To navigate to this page from the My Oracle Support home page, click the Knowledge tab, and then click the Tools and Training menu, JD Edwards EnterpriseOne, Welcome Center, Release Information Overview.

The most current versions of this guide and all other JD Edwards EnterpriseOne Tools and Installation/Upgrade documentation is available on the Oracle Technology Network:

<http://www.oracle.com/technetwork/documentation/jdedent-098169.html>

This guide contains references to server configuration settings. It is highly recommended that you only access and manage these settings for the supported server types using the Server Manager program. For additional details, refer to the *JD Edwards EnterpriseOne Server Manager Guide*.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	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 in examples, text that appears on the screen, or text that you enter.

Understanding the Transaction Server Components

This chapter describes:

- [Section 1.1, "Overview"](#)
- [Section 1.2, "Certifications \(Formerly Known as Minimum Technical Requirements\)"](#)
- [Section 1.3, "Obtaining the Transaction Server Managed Software Component"](#)
- [Section 1.4, "Prerequisites"](#)
- [Section 1.5, "Understanding Transaction Server Communications"](#)

1.1 Overview

The JD Edwards EnterpriseOne Transaction Server Components installer performs the installation, upgrade, and configuration of the Transaction Server Components. These components provide the Real Time Events functionality on the Transaction Server.

This document is not a substitute for the administration manuals provided by your Oracle WebLogic Application Server vendor, the network administration manuals provided by your network vendor, or the installation and configuration manuals for third-party products used with JD Edwards EnterpriseOne.

The procedures in this guide support the Oracle WebLogic Application Server. This document is designed for management information system (MIS) managers and installers. To successfully install the Transaction Server Components on an Application Server, you must have a working knowledge of these topics:

- Hardware and software requirements
- Oracle WebLogic Application Server setup and administration
- JD Edwards EnterpriseOne platforms and operating systems

The recommended method of obtaining this information is to attend the relevant training courses. Information about course offerings, dates, and locations is available on My Oracle Support. At a minimum, read these guides before beginning:

- *JD Edwards EnterpriseOne Tools Foundation Guide*
- *JD Edwards EnterpriseOne Tools Configurable Network Computing Implementation Guide*
- *JD Edwards EnterpriseOne Tools System Administration Guide*
- *JD Edwards EnterpriseOne Tools Package Management Guide*

- *JD Edwards EnterpriseOne Tools Server and Workstation Administration Guide*

1.2 Certifications (Formerly Known as Minimum Technical Requirements)

Customers must conform to the supported platforms for the release as detailed in the JD Edwards EnterpriseOne Certifications (formerly known as 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>

Refer to the Certifications tab on My Oracle Support and search for this product:

- **JD Edwards EnterpriseOne Transaction Server.**

For additional information on using Certifications, refer to this document on My Oracle Support (<https://support.oracle.com>):

- **Certifications FAQ for JD Edwards EnterpriseOne [Article ID 1525328.1]**

1.3 Obtaining the Transaction Server Managed Software Component

You can obtain the Transaction Server Managed Software Component from:

- Update Center
- Change Assistant

The name of the component to download is called *91xTy*, where *x* is the major tools release (for example, 9.1) and *y* is the maintenance release. For example, the Managed Software Component for the Transaction Server for 9.1.3.0 is called 91T30.

Note: Once obtained, the Transaction Server Managed Software Component is installed using the JD Edwards EnterpriseOne tool called Server Manager. Refer to the *JD Edwards EnterpriseOne Server Manager Guide* for additional details.

1.4 Prerequisites

These prerequisites are required prior to installing the Transaction Server Components on an Application Server.

- Oracle WebLogic Application Server must be installed, configured and running. Refer to the JD Edwards EnterpriseOne 9.1.X HTML Server Certifications (Doc ID 1506201.1) for JD Edwards EnterpriseOne-specific installation instructions for the application servers (including patches).
- Verify that your installation meets the prerequisites for installing the Oracle WebLogic Application Server at this link:

<http://www.oracle.com/us/products/middleware/cloud-app-foundation/weblogic/overview/index.html>

1.5 Understanding Transaction Server Communications

The JD Edwards EnterpriseOne Server Manager communicates with the Transaction Server using JMX (Java Management Extensions). The Embedded Agent in the

Transaction Server does not connect to HTTP nor does it connect using Plugins. This is the mechanism by which Server Manager manages the Transaction Server.

The Embedded Agent generally allocates ports in an incremental manner. For example, if a Server Manager Agent installed on the machine is running on port number 14502 (default) then the Embedded Agents running inside managed instances on that machine would allocate ports starting from $14502+1=14503$, 14504, and so on depending on the number of managed instances on that machine.

Tip: In order to ascertain the Port Number on which the Transaction Server Embedded Agent is running, open the <CONTAINER_NAME>.log file of the J2EE Container and search for the string *Management Agent* which should take you to a line containing this string:

```
00000017 Server I Starting the management agent listener on port
'14503'
```

The Transaction Server communicates with the JD Edwards EnterpriseOne Enterprise Servers using the EnterpriseOne Proprietary Communication Protocol; that is, JDENET messages.

The Transaction Server connects to the JD Edwards EnterpriseOne database using proprietary database middleware (JDBj). The default ports are dependent on the database type:

- 1521 for Oracle database
- 1433 for SQL Server database
- 50000 for DB2 database

Although the above are default ports, it is possible to run these databases on different ports. Further, these ports are only defined for the Bootstrap database. The JDBj code retrieves the port numbers for the other databases from the Data Source Master Table based on the port numbers for each of the Data Source records as configured during the Planner and Workbench and the Work with Data Sources Application.

Configuring the Transaction Server Components

You must use Server Manager Console to install and configure Transaction Server instances. These Server Manager settings for the Transaction Server are found under Configuration section of Transaction Server Instance:

- [Section 2.1, "JDBj Database Configuration"](#)
- [Section 2.2, "Real Time Events"](#)
- [Section 2.3, "Network Settings"](#)
- [Section 2.4, "Log File Configuration \(jdelog.properties\)"](#)

2.1 JDBj Database Configuration

These sections of Server Manager control the JDBj settings:

- [Section 2.1.1, "JDBj Bootstrap Datasource"](#)
- [Section 2.1.2, "JDBj Bootstrap Session"](#)
- [Section 2.1.3, "JDBj Connection Pools"](#)
- [Section 2.1.4, "JDBC Drivers"](#)
- [Section 2.1.5, "JDBj Logging"](#)
- [Section 2.1.6, "Oracle Database Settings"](#)
- [Section 2.1.7, "JDBj Runtime Properties"](#)
- [Section 2.1.8, "JDBj Spec Datasource"](#)

2.1.1 JDBj Bootstrap Datasource

This section defines the data source where OCM and some other system tables reside. JDBj uses this at bootstrap time and later to look up OCM entries on demand. If the information entered here is not valid, you cannot access the JD Edwards EnterpriseOne database.

With a Transaction Server instance selected, the following shows typical values:

JDBj Bootstrap Datasource

This section defines the data source where OCM and some other system tables reside. JDBj uses this at bootstrap time and later to look up OCM entries on demand. If it is not valid, then no JD Edwards EnterpriseOne database access is possible.

System Datasource Name	<input type="text" value="System - 910"/>		
Database Type	<input type="text" value="Oracle Database"/>		
Database Name	<input type="text" value="ie911"/>		
Database Server Name	<input type="text" value="den60204jems"/>		
Database TCP/IP Port	<input type="text" value="1613"/>		
Physical Database	<input type="text"/>		
Object Owner	<input type="text" value="SY910"/>		
Supports Large Objects (LOBs)	<input type="checkbox"/>		
Unicode Database	<input type="checkbox"/>		

Below is a list of the settings and typical values:

- *System Datasource Name*
Enter the name of your System datasource. For example:
System - 910
- *Database Type*
Use the pulldown to select your database type.
- *Database Name*
Enter your database name.
- *Database Server Name*
Enter the name of your database server.
- *Database TCP/IP Port*
Enter the TCP/IP port for your database.
- *Physical Database*
Optionally you can choose to specify your physical database.
- *Object Owner*
Enter the object owner for the System datasource. For example:
SY910

2.1.2 JDBj Bootstrap Session

This information is used to sign on with access to system tables. For JD Edwards EnterpriseOne HTML servers, only the user and environment need to be supplied. JDBj signs this user on during the bootstrap process. If the sign on information is not valid, then no access is granted to the JD Edwards EnterpriseOne database. To configure as a Portal WSRP Producer, you must supply a valid EnterpriseOne user and environment and the password and role settings are optional.

With a Transaction Server instance selected, the following shows typical values:

JDBj Bootstrap Session [Return To Top](#)

This information is used to sign on to provide access to system tables. For EnterpriseOne HTML servers, only the user and environment need to be supplied. JDBj signs this user on via the security server during the bootstrap process. If the sign on information is not valid, then no access to the JD Edwards EnterpriseOne database is possible. To configure as a Portal WSRP Producer, a valid E1 user and environment need to be supplied and the password and role settings below are optional.

Bootstrap User

Bootstrap User Password

Bootstrap Role

Bootstrap Environment

[Revert to Defaults](#) [Apply](#)

Below is a list of the settings and typical values:

- *Bootstrap User*
Enter the name of your Bootstrap User. For example:
JDE
- *Bootstrap User*
Enter the password for your Bootstrap User.
- *Bootstrap Role*
Enter the role of your Bootstrap User. For example:
***ALL**
- *Bootstrap Environment*
Enter the bootstrap environment. For example:
JDV910

2.1.3 JDBj Connection Pools

These settings control the database connection pooling (sharing of connections) within JDBj. Connections are pooled at the physical datasource level, not the EnterpriseOne logical datasource level.

With a Transaction Server instance selected, the following shows typical values:

JDBj Connection Pools [Return To Top](#)

These settings control the database connection pooling (sharing of connections) within JDBj. Connections are pooled at the physical data source level, not the EnterpriseOne logical datasource level.

Use JDBC Pooling ☐

Minimum Connections

Maximum Connections

Pool Growth Size

Initial Connections

Connection Timeout

Pool Cleaning Interval

Maximum Prepared Statements

Cache Purge Size

[Revert to Defaults](#) [Apply](#)

Below is a list of the settings and typical values:

- *Minimum Connections*
Enter a value. For example:
0

- *Maximum Connections*
Enter a value. For example:
50
- *Pool Growth Size*
Enter a value. For example:
5
- *Initial Connections*
Enter a value. For example:
5
- *Connection Timeout*
Enter a value in milliseconds. For example:
1800000
- *Pool Cleaning Interval*
Enter a value in milliseconds. For example:
300000
- *Maximum Prepared Statements*
Enter a value. For example:
50
- *Cache Purge Size*
Enter a value. For example:
5

2.1.4 JDBC Drivers

These settings specify which JDBC drivers to use for a particular database type.

With a Transaction Server instance selected, the following shows typical values:

JDBC Drivers [Return To Top](#)

These settings specify which JDBC drivers to use for a particular database type.

Oracle JDBC Driver	?	<input type="text" value="oracle.jdbc.driver.OracleDriver"/>	?
AS/400 JDBC Driver	?	<input type="text" value="com.ibm.as400.access.AS400JDBCDriver"/>	?
SQL Server JDBC Driver	?	<input type="text" value="com.microsoft.sqlserver.jdbc.SQLServerDriver (SQL Server :"/>	?
UDB JDBC Driver	?	<input type="text" value="COM.ibm.db2.jdbc.app.DB2Driver"/>	?

[Revert to Defaults](#) [Apply](#)

Below is a list of the settings and typical values:

- *Oracle JDBC Driver*
Enter a value. For example:
oracle.jdbc.driver.OracleDriver
- *AS/400 JDBC Driver*
Enter a value. For example:

com.ibm.as400.access.AS400JDBCDriver

- *SQL Server JDBC Driver*

Enter a value. For example:

com.microsoft.sqlserver.jdbc.SQL ServerDriver (SQL Server)

- *UDB JDBC Driver*

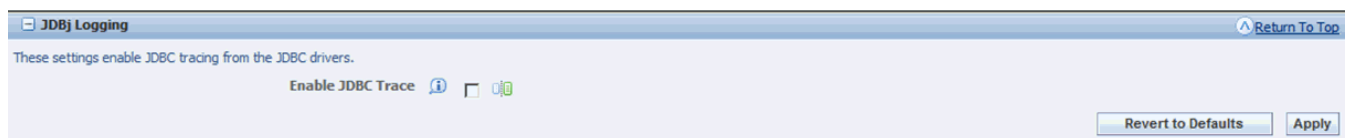
Enter a value. For example:

COM.ibm.db2.jdbc.app.DB2Driver

2.1.5 JDBj Logging

This setting enables JDBC tracing from the JDBC drivers.

With a Transaction Server instance selected, the following shows typical setting:



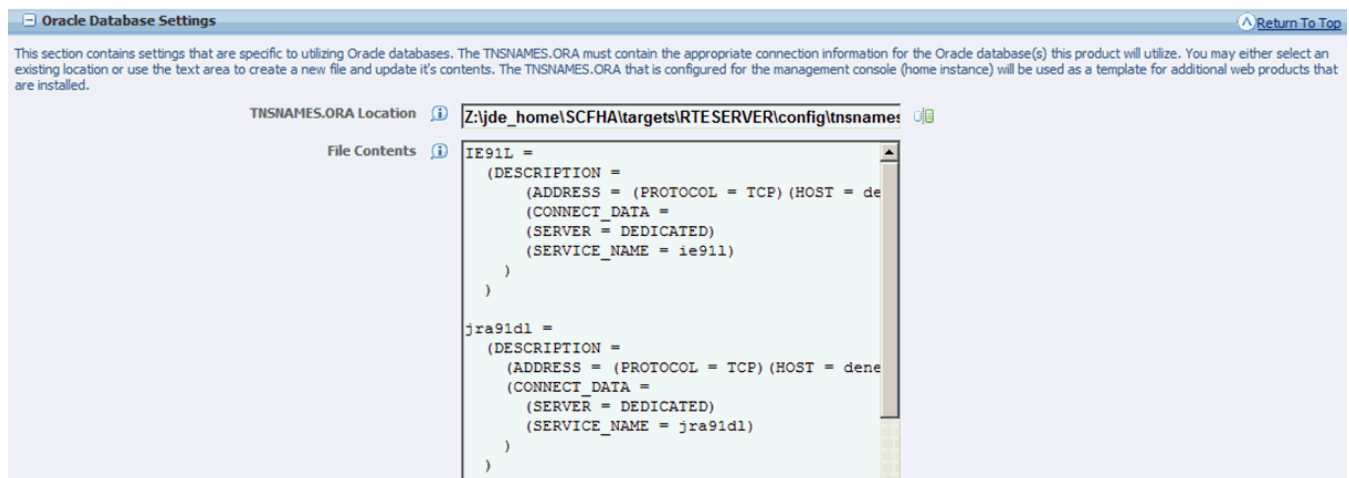
- *Enable JDBC Trace*

Click the checkbox to enable or disable a JDBC trace from the JDBC drivers.

2.1.6 Oracle Database Settings

This section contains settings that are specific to utilizing Oracle databases. The TNSNAMES.ORA must contain the appropriate connection information for the Oracle database(s) this product will utilize. You may either select an existing location or use the text area to create a new file and update its contents. The TNSNAMES.ORA that is configured for the Management Console (home instance) will be used as a template for additional web products that are installed.

With a Transaction Server instance selected, the following shows typical values:



Below is a list of the settings and typical values:

- *TNSNAMES.ORA Location*

Specifies the location of the TNSNAMES.ORA file.

- *File Contents*

Displays the contents of the TNSNAMES.ORA file that will be used by the Server Manager Management Console.

2.1.7 JDBj Runtime Properties

These settings correspond to system runtime properties applicable to JDBj logical connections.

With a Transaction Server instance selected, the following shows the typical value:

Below is the setting:

- *Transaction Timeout*

Specifies a value for the transaction timeout. The default is 0.

2.1.8 JDBj Spec Datasource

This optional section defines the data source where the serialized spec tables reside. JDBj uses these parameters at bootstrap time and after to look up serialized spec objects on demand. If this section is not specified, then JDBj will use the OCM to find the serialized table. If it is not valid, no JD Edwards EnterpriseOne database access is possible.

With a Transaction Server instance selected, the following shows the typical values:

Below are the settings:

- *System Datasource Name*

Specifies a value for the System datasource name.

- *Database Type*

For example, a valid database type is **Oracle Database**.

- *Database Name*

- *Database Server Name*

- *Database TCP/IP Port*
- *Physical Database*
- *Object Owner*
- *Supports Large Objects (LOBs)*
- *Unicode Database*
- *Database User Name*
- *Database Password*

2.2 Real Time Events

This section describes these topics:

- [Section 2.2.1, "Event Processor Configuration Settings"](#)
- [Section 2.2.2, "Trigger Listener Kernel"](#)

2.2.1 Event Processor Configuration Settings

With a Transaction Server instance selected, the following shows the typical values for Event Processor Configuration Settings:

Setting	Value
Sequencing Behavior	Events can be delivered in any sequence.
Processing Concurrency	1
Trigger Listener Delay	10000
Max Transfer Session Size	5000
Initial Context Factory	weblogic.jndi.WLInitialContextFactory
JNDI Provider URL	t3://dnvmiedp12:1014
Event Transfer Transaction Isolation	serializable
Event Client Session Check Interval	60000
Event Client Session Timeout	3600000
App Server JNDI User	*****
App Server JNDI Password	*****

Below are the settings:

- *Sequencing Behavior*

This setting describes how the event sequencing occurs during Transaction Server processing.

Event sequencing behavior values are:

- *AnySequence*
Events can be delivered in any sequence.
- *ByEventType*
Events are guaranteed to be delivered in sequence by event type.
- *ByUserSession*
Events are guaranteed to be delivered in sequence by originating user session.

- *GlobalFIFO*

Events are guaranteed to be delivered in sequence by their original generation sequence. This option does not permit concurrent processing and may impact performance. When this option is selected the **Processing Concurrency** value is ignored.

- *Processing Concurrency*

This setting describes the number of concurrent event processing threads to run. Maximum of 4 concurrent threads can run enhancing the Transaction Server processing performance.

- *Trigger Listener Delay*

This setting specifies the time, in milliseconds, between timer checks of the database for events waiting to be processed. A negative value will disable the timer.

- *Max Transfer Session Size*

This setting specifies the maximum number of events to transfer in one transfer session (that is, one result set from the database).

- *Initial Context Factory*

The InitialContextFactory for the JNDI service of the event server. This value is set as per the transaction server environment.

- *Provider URL*

The Provider URL for the JNDI service of the event server. This value is automatically set as per the transaction server environment.

- *Event Transfer Transaction Isolation*

The transaction isolation level to be used by the transfer agent when reading committed events from the database.

Suggested values based on database server type hosting the F90710 table are shown in the following table:

Server Type	Isolation Level
AS/400	repeatable read
Oracle	serializable
SQL Server	serializable
IBM DB2	repeatable read

- *Event Client Session Check Interval*

This configuration is to check the frequency to check for an expired session.

- *Event Client session Timeout*

This configuration is to check how long a session can be idle before it is timed out.

- *App Server JNDI User*

- *App Server JNDI Password*

2.2.2 Trigger Listener Kernel

This setting applies to the TriggerListener Kernel thread responsible to receive messages from the Enterprise Server that an event has been triggered. All of the configuration settings found under this section are predefined and should **not** be changed.

2.3 Network Settings

This section describes these topics:

- [Section 2.3.1, "JDENET Configuration"](#)
- [Section 2.3.2, "Security Server Configuration"](#)

2.3.1 JDENET Configuration

These settings configure how JDENET, which is the EnterpriseOne messaging protocol, is configured. Any changes to the JDENET configuration will require restarting the server to take effect.

Below are the settings:

- *Outgoing JDENET Port*
- *Back Off Time*
- *Incoming JDENET Port*
- *Kernel Ranges*
- *Timeout Threshold*
- *Pool Size*

2.3.2 Security Server Configuration

These settings configure the EnterpriseOne Enterprise Server to use for security services for this instance. Any changes you make here will not take effect until the server is restarted.

Security Server Configuration

[Return To Top](#)

These settings configure the EnterpriseOne enterprise server to use for security services for this instance. Any changes made will not take effect until the instance is restarted.

Security Server Count

1

Primary Security Server

den60204jems

Secondary Security Server

NONE

Third Security Server

NONE

Fourth Security Server

NONE

Fifth Security Server

NONE

Revert to Defaults

Apply

Below are the settings:

- Security Server Count
- Primary Security Server
- Secondary Security Server
- Third Security Server
- Fourth Security Server
- First Security Server

2.4 Log File Configuration (jdelog.properties)

This section is used to configure logs. You may add, remove, or configure the log files contained within this section. You must click the **Apply** button on this form to save any changes made to this Log Configuration section.

Log File Configuration

[Return To Top](#)

Shown below are the log configurations found in the jdelog.properties for this instance. You may add, remove, or configure the log files contained within. It is necessary to click the **Apply** button below this form to save the changes made to the Log Configuration.

Select [Log Configuration]:

Delete

Select All | Select None

Log Name	Log File Name	Log Level Threshold	Log Format	Append Log Files	Max. Log Size	Max. Backup Index	Log Components
<input type="checkbox"/> E1LOG	Z:\jde_home\SCFHA\targets\RTESERVER\logs\le1root.log	Warnings and Recoverable Errors	Application Format	TRUE	10MB	20	ALL LB

Create New Log Configuration

Apply

Configuring JD Edwards EnterpriseOne

After configuring your Transaction Server, you must set up the Transaction Server Components data for your JD Edwards EnterpriseOne environments to properly generate events.

This chapter discusses:

- [Section 3.1, "Verifying User Defined Codes \(UDCs\)"](#)
- [Section 3.2, "Verifying Tables"](#)
- [Section 3.3, "Verifying Serialized Objects"](#)
- [Section 3.4, "Activating Events"](#)
- [Section 3.5, "Creating Triggers"](#)

3.1 Verifying User Defined Codes (UDCs)

The Transaction Server Components requires the specific User Defined Codes (UDCs) to be populated as shown in this table.

Note: By default, the UDCs are delivered and populated correctly. The information in this section is only provided for verification purposes.

UDC	Purpose
H90/EC	Event Category
H90/TP	Event Transport Parameter Type

3.2 Verifying Tables

The Transaction Server Components requires the use of the tables listed below. The default configuration for RTE is where all of the F907* tables (event tables) reside in one datasource. Out of the box these tables are included in the System datasource database (that is, SY910). Using this default configuration, only one Transaction Server may be configured to monitor the F907* tables. Verify that these tables were created in the JD Edwards EnterpriseOne environments that will generate Real Time Events (RTE) events. Ensure that the server map OCMs and client map OCMs point to the same set of tables.

Table	Description
F90701	Interoperability Event Definition
F90705	Event Activation
F90706	Event Subscriber
F90707	Event Subscription
F907071	Subscribed Events
F907072	Subscribed Environments
F90708	Event Sequence
F90710	Event Transfer
F90711	Event Transport Parameters
F90712	Event Transfer Failure
F90715	Event Subscriber Queues
F90720	Event Metrics
F90730	Unique Sequence

3.3 Verifying Serialized Objects

Serialized objects for tables, specs, and data structures must exist on the database machine. Verify that the serialized objects have been generated and copied to the database machine. For more information on performing the generation, see the *JD Edwards EnterpriseOne HTML Web Server Reference Guide*, specific for your platform, and the *Generating JD Edwards EnterpriseOne Serialized Objects* chapter.

3.4 Activating Events

Using this procedure, you must use JD Edwards EnterpriseOne universal batch engine (UBE) processes to convert the existing event data to be compatible with the new tables.

1. Run the Populate Event Activation Status Table (R90705) to create an activation record for every environment for every event defined in the F90701 table. Through use of the processing options you can process this UBE for all events and all environments or for a single event and/or environment. You can run this UBE multiple times if necessary.

Processing Options

Create Event Activation Rules

Specify the event name to create records for:

Note: Leaving this field blank will create records for all events. If this field is not blank, records will only be created for the specified event name regardless of what is specified in the event type specification.

Specify the interoperability event type to create records for:

Note: Leaving this field blank will create records for all event types.

Specify the release of the environments that records are to be created for:

Note: Leaving this field blank will create records for all environments.

OK **Help** **Cancel**

2. On Processing Options, Create Event Activation Rules, complete these fields:

Field	Description
<i>Specify the event name to create records for:</i>	If you leave this field blank, records are created for all events. If you specify a value for this field, records are only created for the specified event name regardless of what is specified in the event type specification.
<i>Specify the interoperability event type to create records for:</i>	<p>If you leave this field blank, records are created for all event types. Valid event types are:</p> <ul style="list-style-type: none"> ■ RTE ■ WF ■ XAPI ■ ZFILE
<i>Specify the release of the environments that records are to be create for:</i>	If you leave this field blank, records are created for all environments. Values are supported releases of JD Edwards EnterpriseOne, specified by internal designations.

Note: You may use data selection to select on events that are currently active in the event system to convert.

3. Run the Convert Event Subscriptions (R90706) UBE to populate the Queue Assignment Table (F90715) with the subscriber queues defined in the Transaction

Server. You can use processing options to initially create your subscriber queue records and convert your subscribers and subscriptions. Later if you add new subscribers you can use this UBE to only create the new subscriber queue records. Initially there are 16 subscriber queues delivered with JD Edwards EnterpriseOne.

Note: The R90706 program does not create the subscriber queues. Instead, 16 of these subscriber queues were created during the initial installation of the Transaction Server Components on an Application Server. If you have more than 16 subscribers you will need to create the additional queues using Application Server administrator tools prior to running this program.

Processing Options

Conversion Parameters

Enter the number of subscriber queues to create:

Enter the description that will be assigned to all converted subscriptions. If this field is left blank the default description will be "Subscription converted by R90706".

Do not convert subscriptions:

Enter a '1' to prevent batch application from converting subscribers and subscriptions. If value is '1' then batch application will only create the number of subscriber queues specified.

OK **Help** **Cancel**

4. On the Conversion Parameters tab, complete these fields:

Field	Description
<i>Enter the number of subscriber queues to create:</i>	Set this value to the number of subscriber queues for your installation. You must create a queue for every subscriber.
<i>Enter the description that will be assigned to all converted subscriptions. If this field is left blank the default description will be "Subscription converted by R90706".</i>	Enter a value to describe all converted description. If you leave this field blank the default description is "Subscription converted by R90706".

Field	Description
<i>Do not convert subscriptions</i>	<p>Values are:</p> <p>0: Convert subscriptions</p> <p>1: Do not convert subscribers and subscriptions. The R90706 program only creates the number of subscriber queues that you specify and does not perform any conversions.</p>

Note: You may use data selection to select on events that are currently active in the event system to convert.

- Run the Synch Z Events in F47002 with F90701 (R90701A) UBE to create event records in the F90701 for the z-event records in the F47002 table for records that do not currently exist in the F90701 table. There are no processing options for this UBE.

3.5 Creating Triggers

You must create a trigger on the F90710 table. Prior to running the trigger creation script, ensure that you have adequate database privileges required to create a trigger. This table lists the privileges by database.

Note: Additionally, as described in the Prerequisites sections of this guide, you must obtain and install a specific ESU to enable the creation of triggers for Transaction Server Components.

Database	Required Privilege
IBM DB2 for System i	No special privileges are required.
IBM DB2	<p>Grant the following privileges to the database proxy user. Replace <i>jde</i> in the following script with the actual database proxy user name.</p> <pre>GRANT CONTROL ON TABLE SY900.F90710 TO USER jde</pre>
Oracle	<p>Grant the following privileges to the JDE_ROLE or the database proxy user.</p> <pre>GRANT CREATE ANY SEQUENCE TO JDE_ROLE; GRANT DROP ANY SEQUENCE TO JDE_ROLE; GRANT CREATE ANY TRIGGER TO JDE_ROLE; GRANT DROP ANY SEQUENCE TO JDE_ROLE;</pre>
SQL Server	<p>The database proxy user must have the privileges to create database triggers. Replace <i>jde</i> with the actual database proxy user name.</p> <pre>Exec sp_addrolemember db_owner, jde</pre>

You must create the (F90710) Event Transfer and the (F90730) Unique Sequence in the same JD Edwards EnterpriseOne data source. You should verify this configuration using OCM.

- Run this script from a JD Edwards EnterpriseOne Microsoft Windows-based Web Development Client:

```
x:\JDEdwards\E900\system\bin32\dbtemplates.exe -create
```

2. After the scripts complete successfully, you should revoke any special privileges you granted in order to create the triggers. This table lists the revocation commands by database.

Database	Required Privilege
IBM DB2 for System i	No special privileges are required.
IBM DB2	Revoke the following privileges to the database proxy user. Replace <code>jde</code> in the following script with the actual database proxy user name. <pre>REVOKE CONTROL ON TABLE SY900.F90710 TO USER jde</pre>
Oracle	Revoke the following privileges to the <code>JDE_ROLE</code> or the database proxy user. <pre>REVOKE CREATE ANY SEQUENCE TO JDE_ROLE; REVOKE DROP ANY SEQUENCE TO JDE_ROLE; REVOKE CREATE ANY TRIGGER TO JDE_ROLE; REVOKE DROP ANY SEQUENCE TO JDE_ROLE;</pre>
SQL Server	Replace <code>jde</code> with the actual database proxy user name. <pre>exec sp_droprolemember db_owner, jde</pre>

Adding a Subscriber

This chapter discusses these tasks:

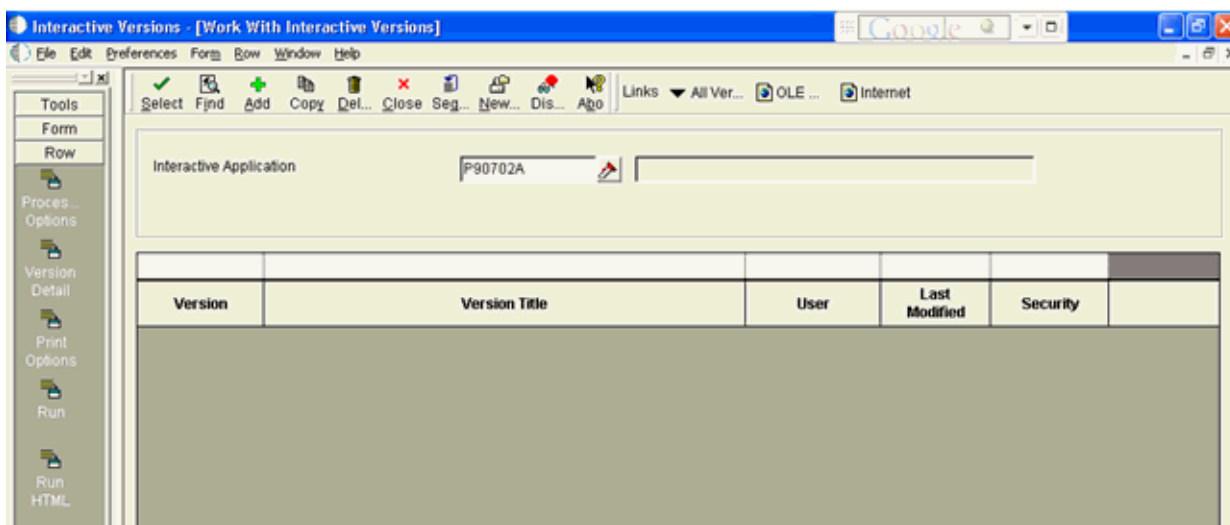
- [Section 4.1, "Setting Processing Options for Event Subscribers Application \(P90702A\)"](#)
- [Section 4.2, "Using Event Subscribers Application to Add a Subscriber"](#)
- [Section 4.3, "Working with Subscribers in JD Edwards EnterpriseOne"](#)

4.1 Setting Processing Options for Event Subscribers Application (P90702A)

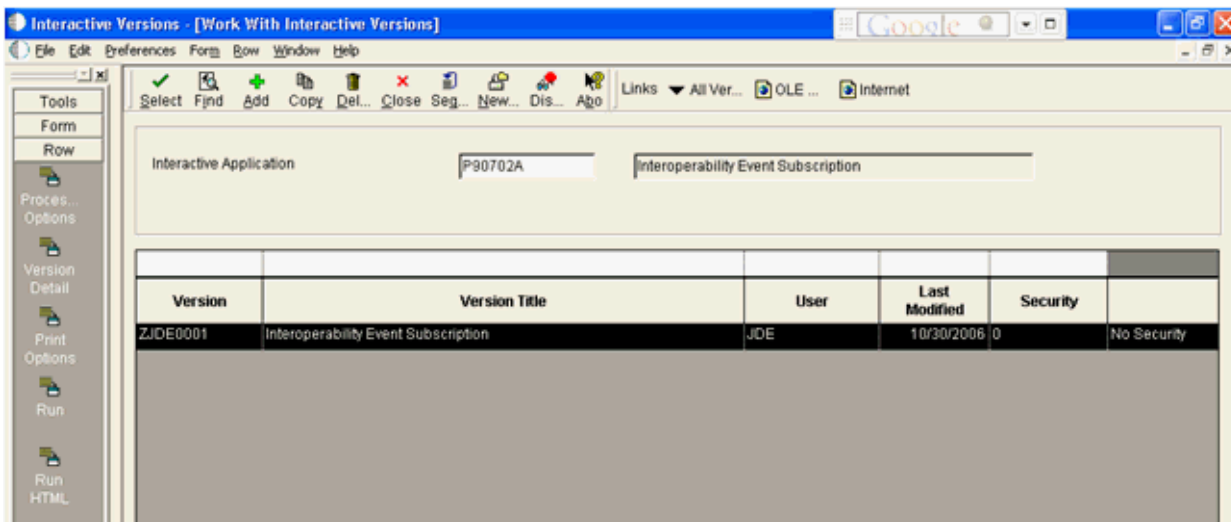
This section discusses how to set up the default values for the processing option.

To set Processing Options for P90702A:

1. Type IV in the *Fast Path*.
2. Type P90702A in the *Interactive Application* and click *Find*.



3. Select the row that appears in the grid and then click *Processing Options* from *Row* menu.



- On *Processing Options*, edit the default values and click OK.

JMSQUEUE	
Oracle Application Server Initial Context Factory	com.evermind.server.rmi
Oracle Application Server Local Provider URL	ormi://localhost:23791
Oracle Application Server Remote Provider URL	ormi://remote-machine-n
Websphere Initial Context Factory	com.ibm.websphere.nam
Websphere Local Provider URL	corbaloc:iiop:localhost:2
Websphere Remote Provider URL	corbaloc:remote-machin

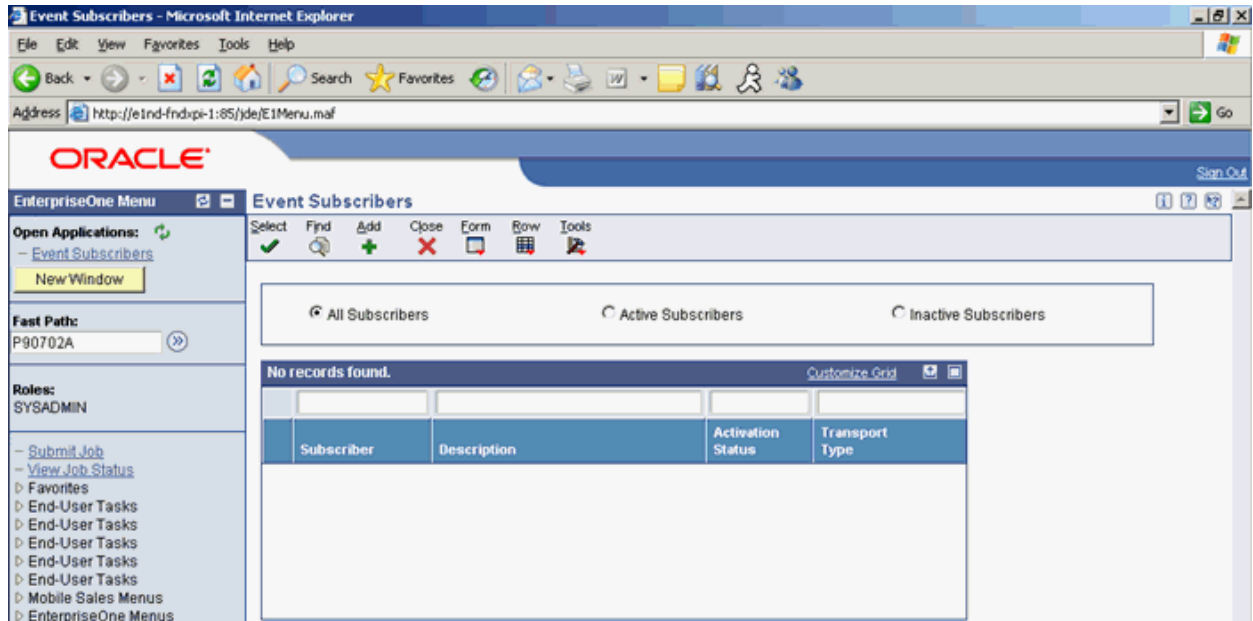
Note: The values for these processing options are set by using an EnterpriseOne application as described in the next section of this chapter in the section entitled: [Section 4.2, "Using Event Subscribers Application to Add a Subscriber"](#).

4.2 Using Event Subscribers Application to Add a Subscriber

This section discusses how to add a subscriber to EnterpriseOne by executing the application P90702A.

To use P90702A to add a JMSQUEUE or JMSTOPIC subscriber:

1. Run the application P90702A from JAS.



2. Click the *Add* button. Enter the values as per any one of the following options:

The Initial Context Factory and Provider URL values are populated based on the selections for Application Server and Queue Location. Edit these values according to the environment. A processing option determines the default values.

The values for **Connection Factory JNDI Name** and **Queue Name** are the JNDI name for the subscriber Queue or Topic that is being created.

4.3 Working with Subscribers in JD Edwards EnterpriseOne

This section describes these tasks:

- [Section 4.3.1, "Adding a JAVACONN Subscriber"](#)
- [Section 4.3.2, "Creating a JMSQUEUE Subscriber"](#)
- [Section 4.3.3, "Creating a JMSTOPIC Subscriber"](#)
- [Section 4.3.4, "Creating a Subscription"](#)
- [Section 4.3.5, "Adding an Event to the Subscription"](#)
- [Section 4.3.6, "Adding Environment\(s\) to a Subscription"](#)

4.3.1 Adding a JAVACONN Subscriber

Use this procedure to add a JAVACONN subscriber in JD Edwards EnterpriseOne:

1. From a JD Edwards EnterpriseOne client, run P90702A to use the Add Event Subscriber application to add a subscriber.

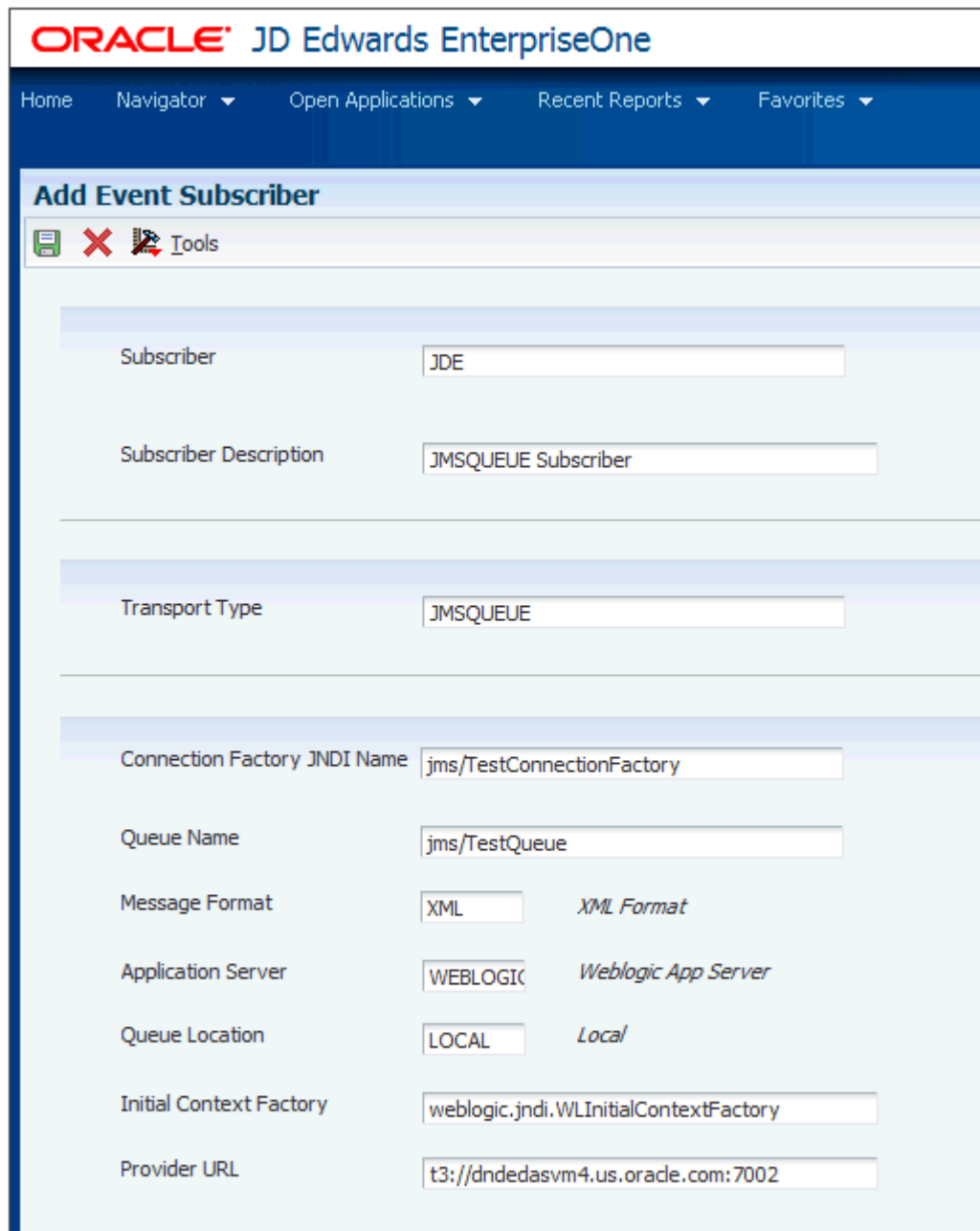
Field	Description
<i>Subscriber</i>	The <i>Subscriber</i> field must have a value matching valid user in JD Edwards EnterpriseOne.
<i>Subscriber Description</i>	Enter this value: JAVACONN Subscriber
<i>Transport Type</i>	Enter this value:. JAVACONN

2. Click the **OK** button to accept the values.
3. Run the P90702A application and find the JAVACONN Subscriber created in the above step.
4. Activate the JAVACONN Subscriber by taking this Form Exit:
Row > Change Status
5. Change the status to AV (Active).

4.3.2 Creating a JMSQUEUE Subscriber

To create a JMSQUEUE subscriber:

1. From a JD Edwards EnterpriseOne client, run P90702A to use the Add Event Subscriber application to add a subscriber.



ORACLE JD Edwards EnterpriseOne

Home Navigator Open Applications Recent Reports Favorites

Add Event Subscriber

Tools

Subscriber: JDE

Subscriber Description: JMSQUEUE Subscriber

Transport Type: JMSQUEUE

Connection Factory JNDI Name:.jms/TestConnectionFactory

Queue Name:.jms/TestQueue

Message Format: XML *XML Format*

Application Server: WEBLOGIC *Weblogic App Server*

Queue Location: LOCAL *Local*

Initial Context Factory: weblogic.jndi.WLInitialContextFactory

Provider URL: t3://dndedasvm4.us.oracle.com:7002

Field	Description
<i>Subscriber</i>	The <i>Subscriber</i> field must have a value matching valid user in JD Edwards EnterpriseOne.
<i>Subscriber Description</i>	Enter this value: JMSQUEUE Subscriber
<i>Transport Type</i>	Enter this value:. JMSQUEUE

Field	Description
<i>Connection Factory JNDI</i>	<p>You can specify any path and name.</p> <p>Note: You must use this same value when creating the Connection Factory in WebLogic as described later in this guide. For purposes of this documentation, the value is:</p> <p>jms/TestConnectionFactory</p>
<i>Queue Location</i>	<p>You can choose either LOCAL or REMOTE.</p> <p>Note: If the subscriber queue is created on the Transaction Server instance you should specify the value as LOCAL. If the subscriber queue is on remote WebLogic container, specify the value as REMOTE</p>
<i>Initial Context Factory</i>	<p>Enter this value:</p> <p>weblogic.jndi.WLInitialContextFactory</p>
<i>Provider URL</i>	<p>Enter this value:</p> <p>t3://<WLS_SERVER_HOST>:<WLS_SERVER_PORT></p> <p>Note: < WLS_SERVER_HOST > is the host of the machine where the subscriber queue is created and <WLS_SERVER_PORT> is the port at which the server is hosted.</p>

2. Click the **OK** button to accept the values.
3. Run the P90702A application and find the JMSQUEUE Subscriber created in the above step.
4. Activate the JMSQUEUE Subscriber by taking this Form Exit:
Row > Change Status
5. Change the status to AV (Active).

4.3.3 Creating a JMSTOPIC Subscriber

To create a JMSTOPIC subscriber:

1. From a JD Edwards EnterpriseOne client, run P90702A to use the Add Event Subscriber application to add a subscriber.

ORACLE JD Edwards EnterpriseOne

Home Navigator Open Applications Recent Reports Favorites

Add Event Subscriber

Tools

Subscriber: JDE

Subscriber Description: JMSTOPIC Subscriber

Transport Type: JMSTOPIC

Connection Factory JNDI Name: jms/TestConnectionFactory

JMS Topic: jms/TestTopic

Field	Description
<i>Subscriber</i>	The <i>Subscriber</i> field must have a value matching valid user in JD Edwards EnterpriseOne.
<i>Subscriber Description</i>	Enter this value: JMSTOPIC Subscriber
<i>Transport Type</i>	Enter this value:. JMSTOPIC
<i>Connection Factory JNDI</i>	You can specify any path and name. Note: You must use this same value when creating the Connection Factory in WebLogic as described later in this guide. For purposes of this documentation, the value is: jms/TestConnectionFactory

- Click the **OK** button to accept the values.
- Run the P90702A application and find the JMSTOPIC Subscriber created in the above step.
- Activate the JMSTOPIC Subscriber by taking this Form Exit:

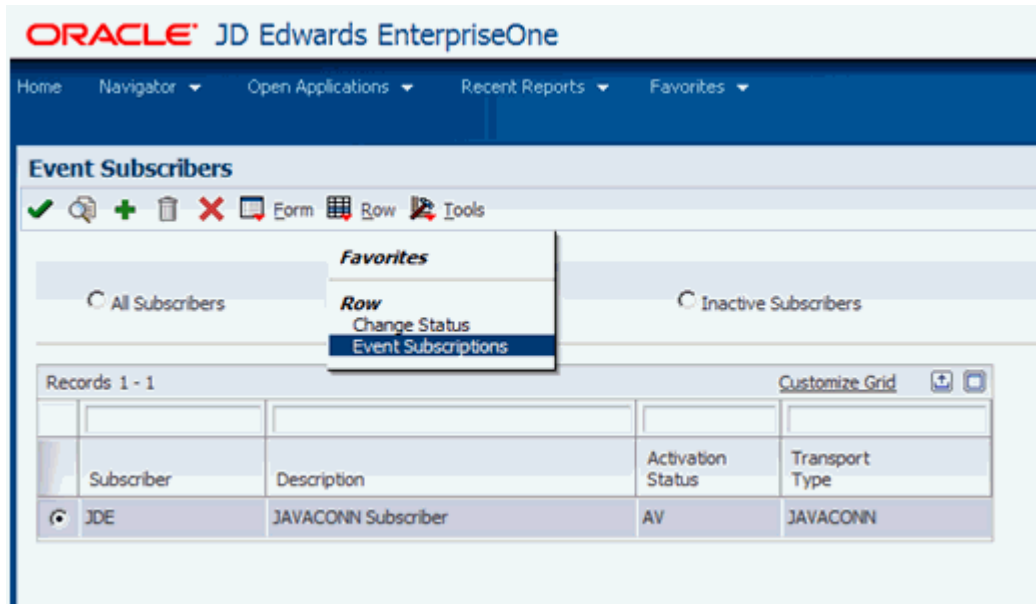
Row > Change Status

5. Change the status to AV (Active).

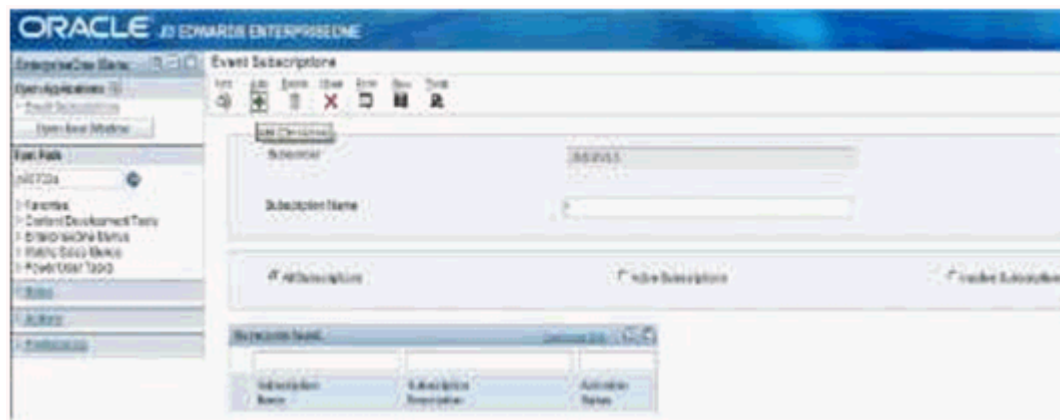
4.3.4 Creating a Subscription

Use this procedure to create a Subscription.

1. From a JD Edwards EnterpriseOne client, run the P90702A application and find the Subscribers you created in the previous steps of this section.



2. On Event Subscribers, take the Row Exit for Event Subscriptions.



3. On Event Subscriptions, click the **Add** button.

Home Navigator Open Applications Recent Reports Favorites

Add Event Subscription

Save Cancel Help Tools

Subscriber JDE

Subscription Name JDE

Subscription Description JDE Subscriber

4. On Add Event Subscription, complete the *Subscriber Name* and *Subscriber Description* fields with any value. In this example, the both values are **JDE**.
5. Click the **OK** button to accept the values.
6. Activate the Subscriber by taking this Form Exit:
Row > Change Status
7. Change the status to AV (Active).

4.3.5 Adding an Event to the Subscription

Use this procedure to add an Event to the Subscription.

Note: In this example, the added event is named RTABOUT.

1. Run P90702A and find the Subscriber created in the previous step of this section.

Note: In this documentation, that subscription was called **JDE**.

ORACLE® JD Edwards EnterpriseOne

Home Navigator Open Applications Recent Reports Favorites

Event Subscriptions

Subscriber

Subscription Name

Favorites
Row
 Subscribed Events
 Subscribed Env
 Change Status

☒ All Subscriptions
 ☐ Active Subscriptions
 ☐ Inactive Subscriptions

Records 1 - 1 [Customize Grid](#)

	Subscription Name	Subscription Description	Activation Status
<input checked="" type="radio"/>	JDE	JDE Subscriber	AV

- On Event Subscriptions, select the *Subscribed Events* Row Exit.

ORACLE® JD Edwards EnterpriseOne

Home Navigator Open Applications Recent Reports Favorites

Subscribed Events

Subscriber

Subscription Name

Records 1 - 2 [Customize Grid](#)

	Event Type	Event Category
<input checked="" type="radio"/>	RTABOUT	RTE
<input type="radio"/>		

- On Subscribed Events, add *RTABOUT* as an *Event Type* by clicking the radio button next to the value.

EnterpriseOne Menu

Open Applications

Subscribed Events

OK Find Delete Cancel Tools

Subscriber JMSQWLS

Subscription Name JMSQWLS

Records 1 - 2

Event Type	Event Category
RTABOUT	RTE

- Click the **OK** button to add *RTABOUT* as a Subscribed Event to the Subscription.

4.3.6 Adding Environment(s) to a Subscription

Use this procedure to add an Environment to a Subscription.

- Run P90702A and find the Subscriber you created earlier in this section of the guide.

ORACLE JD Edwards EnterpriseOne

Home Navigator Open Applications Recent Reports Favorites

Event Subscriptions

Form Row Tools

Subscriber JDE

Subscription Name

Subscribed Env

All Subscriptions Active Subscriptions Inactive Subscriptions

Records 1 - 1

Subscription Name	Subscription Description	Activation Status
JDE	JDE Subscriber	AV

- ## ORACLE® JD Edwards EnterpriseOne

Home Navigator ▼ Open Applications ▼ Recent Reports ▼ Favorites ▼

Subscribed Environments






Tools

Subscriber

JDE

Subscription Name

JDE

Records 1 - 3
[Customize Grid](#)





		Environment	
<input type="radio"/>		DV910	
<input checked="" type="radio"/>		JDV910	
<input type="radio"/>			
<input type="radio"/>			

Troubleshooting Real Time Events Processing

This chapter discusses troubleshooting Real Time Events processing.

Problem: There are some events in the table F90710 which are not getting picked up by the RTE server.

Resolution: Check if the event sequence number stored in the table F90708 is greater than lowest event sequence number available in table F90710. In this case the event sequence number in table F90708 must be reset to 0 manually

Problem: Some of the clustered servers are not processing events.

Resolution: Probably some servers in the cluster are having some problem. Refer the app server log to identify the problem, rectify the problem and restart the cluster.

Resolution: There could be very high number of events processed exceeding the filestore size. It is advisable to consume events while they are being produced and processed. This keeps the filestore clean.

Problem: Getting error in the RTE Server on WAS (Failed to send committed event message: CWSIA0053E: An exception was received during the call to the method JmsSessionImpl.getTransaction (#1): javax.resource.spi.IllegalStateException: CWSJR1121E: An internal error has occurred. During the call to the method getManagedConnection the exception javax.resource.spi.ResourceAllocationException: CWSJR1028E: An internal error has occurred. The exception com.ibm.ws.sib.processor.exceptions.SIMPConnectionUnavailableException)

Resolution: One of the messaging engines is down and unable to process event. This should not be a concern since the failover service is active and event will be processed by other available messaging engines, assuming they are available.

Problem: Error message "JNDI lookup of Queue failed" seen in RTE server log.

Resolution: It is likely that the JNDI resources are not loaded yet. This error most commonly occurs when auto migration is enabled. A retry lookup feature exists which will try to lookup until successful. If the problem persists, then check the application server log

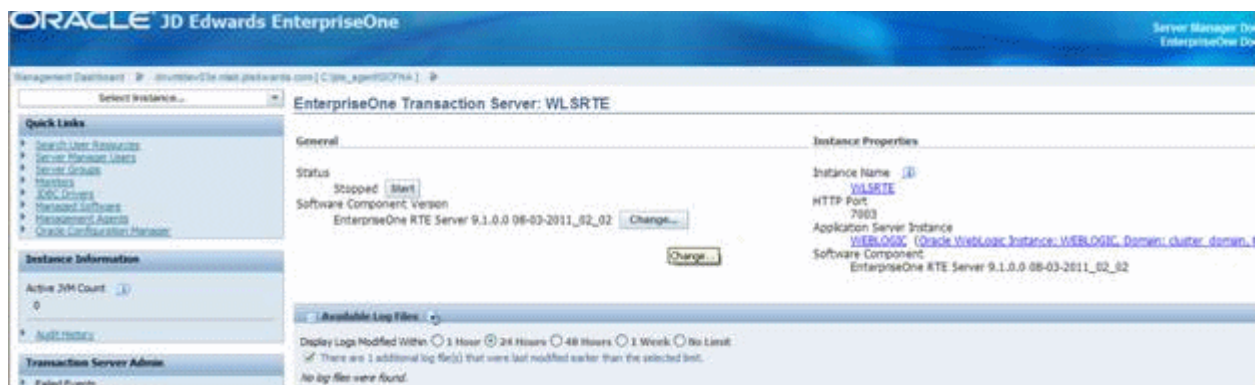
Problem: All the messages are being sent to a single queue in a cluster or not evenly distributed.

Resolution: A failover may have occurred and the Queue configuration has been changed due to failover. Stop the clustered servers, modify the configuration as stated in this document and start the cluster to process events.

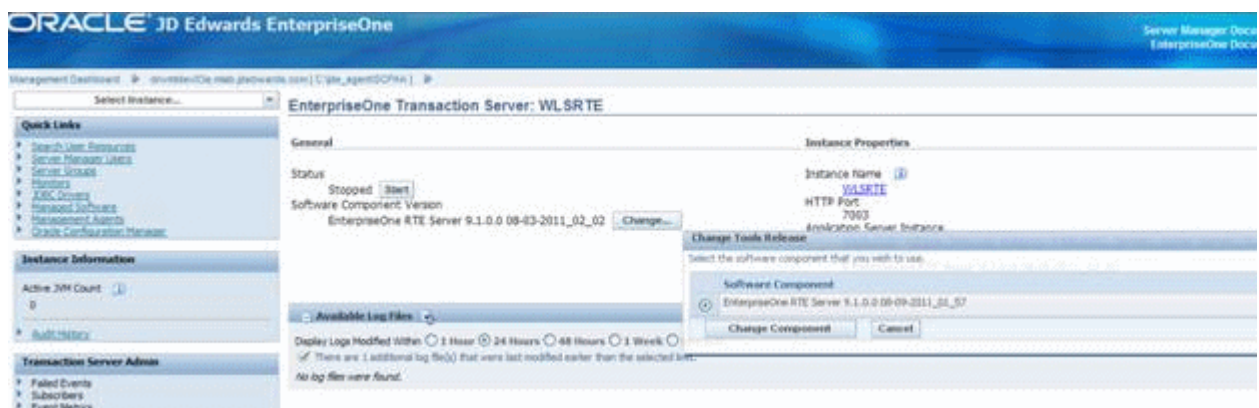
Upgrading the Transaction Server RTE Software Component Using Server Manager

Use this procedure to upgrade the Transaction Server RTE component that was installed using Server Manager.

1. Obtain the RTE software component to which you wish to upgrade from Oracle eDelivery.
2. Place the new RTE component into this folder:
<AGENT_LOC>\components
3. Log on to the Server Manager console.
4. Click on the RTE instance.



5. In the Software Component Version section, click the **Change** button.



-
6. On the Change Tools Release layout, select the desired version to install from the component by selecting the radio button.
 7. Click the **Change Component** button.

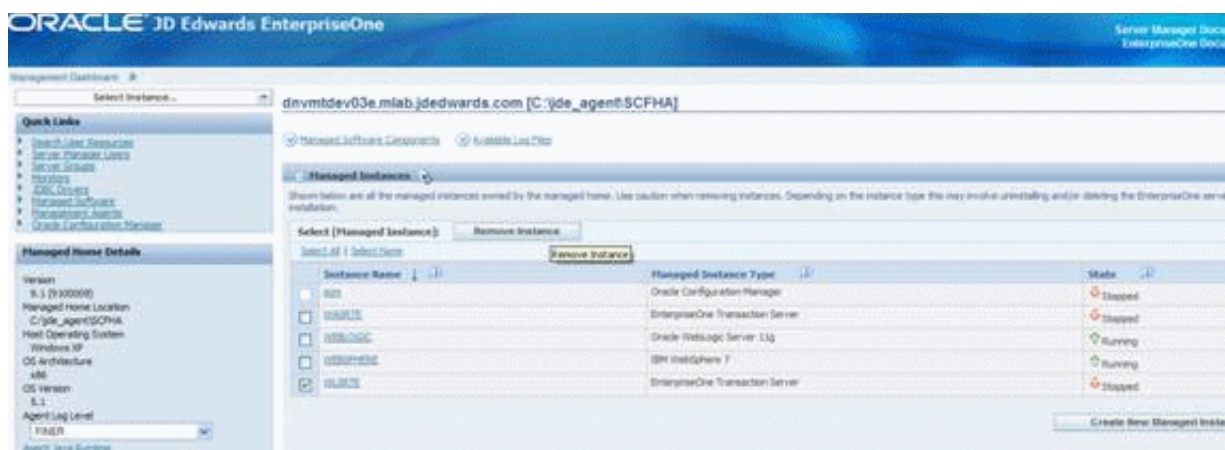
Note: An upgrade of RTE server only upgrades the Transaction Server RTE component of JD Edwards, and not the JMS components.

Refer to the *JD Edwards EnterpriseOne Server Manager Guide* to upgrade the Server Manager Agent. You can only update the Server Manager Agent after you have upgraded the Server Manager Console to Tools Release 9.1.

Uninstalling the Transaction Server RTE Software Component using Server Manager

Use this procedure to uninstall the Transaction Server RTE Software Component using Server Manager.

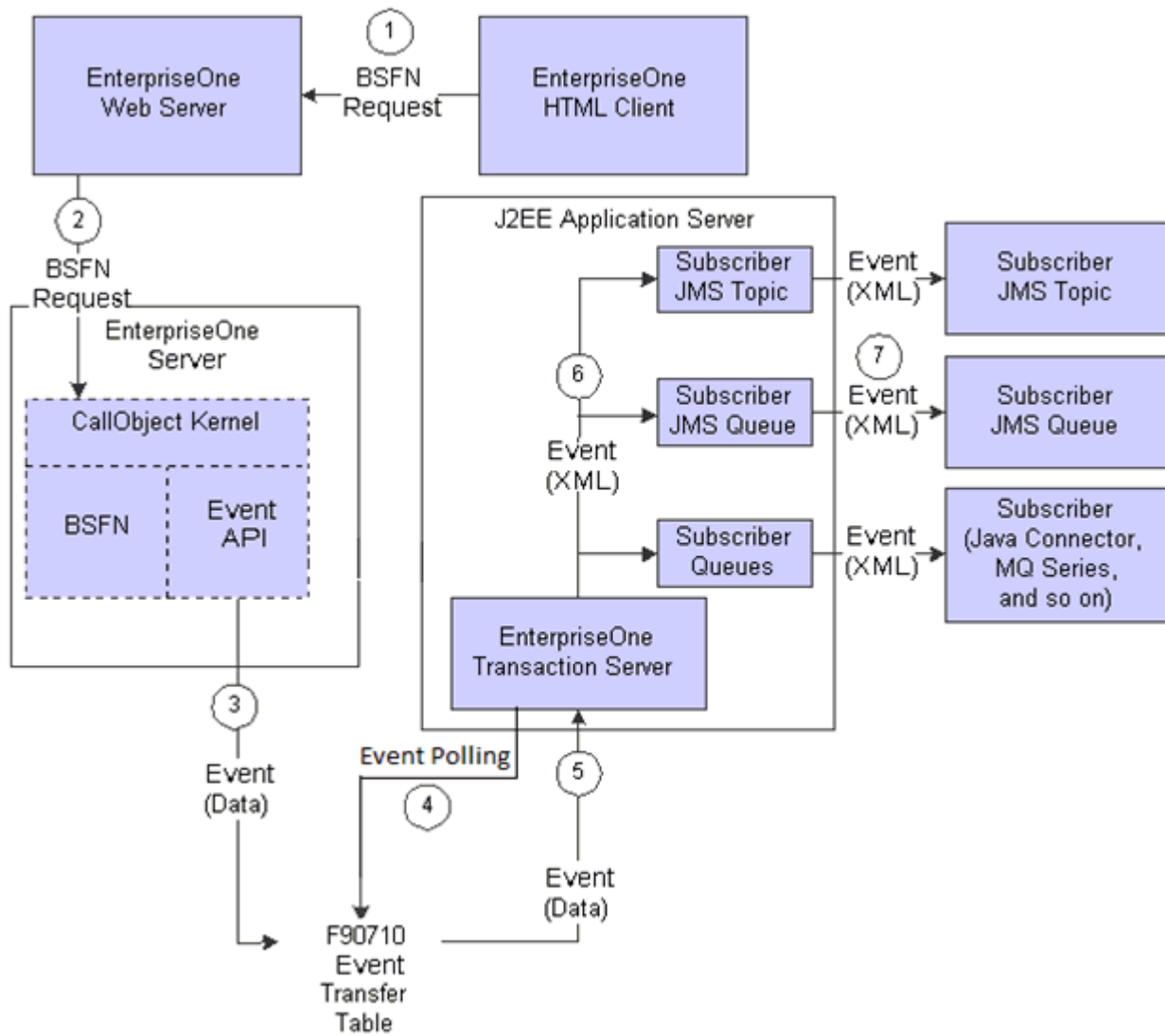
1. Log on to the Server Manager console.
2. Click on the managed home location.
3. Select the Transaction Server Real Time Events (RTE) instance.
4. Click the **Remove Instance** button.



Architecture of the Guaranteed Event System

This appendix presents an architectural diagram of the Guaranteed Event System, which is one of the JD Edwards EnterpriseOne Transaction Server Components. The diagram depicts the general sequence that occurs when an event is published. These steps correspond to the numbered sequence of events in the diagram:

1. An HTML client user executes a business function (BSFN) request.
This request is sent to the JD Edwards EnterpriseOne HTML Web Server.
2. The HTML Web Server forwards this request to a CallObject kernel on the Enterprise Server.
3. The CallObject kernel executes the BSFN, which calls the Event API to send the event data to the F90710 Event Transfer table.
If the event is a Z event, the data is sent to F90710 in a final XML format.
4. A database polling application available with the JD Edwards EnterpriseOne Transaction Server polls any available records in F90710.
5. The transaction server retrieves the event data from F90710 and (for Real-Time and XAPI events) converts the data to an XML document in the appropriate format.
6. The transaction server routes the event to the subscriber queue(s) for each subscriber that has established an active subscription for that event.
7. When a subscriber connects to the Transaction Server, the subscriber is sent all the events that exist in its subscription queue at that time.



Creating and Configuring Multiple Transaction Servers per Environment or User/Role

The default configuration for the Transaction Server defines the F907* tables (event tables) to reside in one datasource. These tables are included in the System datasource database (for example, SY812 or SY900). Using this default configuration, only a single Transaction Server may be configured to monitor the F907* tables.

This appendix provides instructions to set up and configure multiple Transaction Servers so that transactions can be separated by one or more of these criteria:

- Environment
- Role/User

This process allows you to:

- Separate transaction events by environment so that two different transaction servers can service the events
- Use OCM mappings to point to the different environments

This appendix describes these tasks:

- [Section B.1, "Creating and Configuring Multiple Transaction Servers per Environment"](#)
- [Section B.2, "Creating and Configuring Multiple Transaction Servers per Role/User"](#)

B.1 Creating and Configuring Multiple Transaction Servers per Environment

These general guidelines apply:

- *Production Environments*

The F907* tables should remain in the System database as defined by the standard setup and configuration. Since the transaction server is designed to work with only one set of F907* tables, the normal procedure is to configure the transaction server with these system tables.

- *Test and Dev Environments*

In order to separate Test and Dev environment transactions from Production, you can copy the F907* tables into the business data database and change the OCM mappings accordingly. A second transaction server is necessary in order to access

these copied tables because, by design, each Transaction Server can only work with only one set of F907* tables.

- *Multiple Transaction Servers*

In order to support the separation of the F907* tables per environment, you must install an additional Transaction Server per environment or environment combination. For example, if you want to isolate your Production environment and then combine your Test and Dev environments, you would need a total of two Transaction Servers. If you further wanted to separate your Test and Dev environments, you would need a total of three Transaction Servers.

Do not start any added Transaction Servers until you complete the procedure in this appendix.

To create and configure the Transaction Server per environment:

1. With the exception of the tables listed in the *Exceptions* below, copy all remaining F907* tables from the System database to the business data database.

Exceptions

The following tables are initially cached on JD Edwards EnterpriseOne service startup and contain the necessary information needed for all environments and subscribers. Therefore, these tables and their OCM mappings should remain in the System datasource:

- F90701
 - F907011
 - F90705
2. Use OCM to remap the F907* tables copied in Step 1 to the new business data database for both the system and the server map.

For example, the completed mappings might look like these:

```
JPY900 F90701  TBLE  System - 900 *PUBLIC AV
JPY900 F907011 TBLE  System - 900 *PUBLIC AV
JPY900 F907012 TBLE  System - 900 *PUBLIC AV
JPY900 F90703  TBLE  Business Data - CRP *PUBLIC AV
JPY900 F90704  TBLE  Business Data - CRP *PUBLIC AV
JPY900 F90705  TBLE  System - 900 *PUBLIC AV
JPY900 F90706  TBLE  Business Data - CRP *PUBLIC AV
JPY900 F90707  TBLE  Business Data - CRP *PUBLIC AV
JPY900 F907071 TBLE  Business Data - CRP *PUBLIC AV
JPY900 F907072 TBLE  Business Data - CRP *PUBLIC AV
JPY900 F90708  TBLE  Business Data - CRP *PUBLIC AV
JPY900 F90710  TBLE  Business Data - CRP *PUBLIC AV
JPY900 F90711  TBLE  Business Data - CRP *PUBLIC AV
JPY900 F90712  TBLE  Business Data - CRP *PUBLIC AV
JPY900 F90715  TBLE  Business Data - CRP *PUBLIC AV
JPY900 F90720  TBLE  Business Data - CRP *PUBLIC AV
JPY900 F90730  TBLE  Business Data - CRP *PUBLIC AV
```

3. Activate the RT* event for added environment (for example, JPY900) in P90701 by:
 - Taking the Form exit to Event Activation
 - Doing an Add to add the new Event per environment
 - Using the Row exit to change the status to Active

4. Refresh the JD Edwards EnterpriseOne server cache by restarting the EnterpriseOne services.
5. Add subscriber information for the added environment (for example, JPY900).
6. Refresh the Transaction Server cache by restarting the transaction server.
7. Trigger the event.
8. Verify the event is recorded in the F90710 Business Data Source table.
9. From a JD Edwards EnterpriseOne client machine, use this command to run the Table Trigger script for F90710 in the business data for the added environment (for example, PY Business Data):

```
<home>\system\bin32\dbtemplates.exe -create
```

where <home> is the installation directory of the JD Edwards EnterpriseOne Administration client that is named according to the JD Edwards EnterpriseOne release. For example, E812 or E900.

10. Verify the F986112 table is populated and the new event in F90710 has the status 3 instead of 2.

B.2 Creating and Configuring Multiple Transaction Servers per Role/User

This section describes a configuration where multiple Real Time Events can be processed by multiple Transaction Server instances within the same JD Edwards EnterpriseOne environment. You can process events by Role/User to enable specific users to have their events processed by a specific Transaction Server. This setup is similar to that in the previous section, [Section B.1, "Creating and Configuring Multiple Transaction Servers per Environment"](#). Instead of separating the events by environment you are separating the Real Time Event messages by Role/User.

Multiple Transaction Servers

In order to support the separation of the F907* tables per Role/User, you must install an additional Transaction Server per User or Role combination. For example, you have set up a role for processing Sales Orders and another role for processing all other Real Time Events. In this case, you would need an additional Transaction Server for each role addition.

To configure the system to route Real Time Events to specific Transaction Servers based on the JD Edwards EnterpriseOne Role/User user that is generating the event:

1. From JD Edwards EnterpriseOne, create a new Role which will include a group of users that will initiate the Real Time Event messages.
2. Create these tables in the same environment datasource (Business Data - PROD) or create a custom Data Source (for example, TS2 / PD900):
 - F90710 - Event table
 - F90730 - Unique sequence table
3. Copy these tables from the System datasource to the same environment datasource you used in Step 2:
 - F90706 - Event Subscriber
 - F90707 - Event Subscription
 - F907071 - Subscribed Events
 - F907072 - Subscribed Environments

- F90708 - Event Sequence

These tables contain the subscriber information and whether they are active or not. It is important to note that when you copy these tables, whenever you want to make changes to the subscriber information you must ensure you are logged in as the appropriate user. This is required to ensure you are making changes to the correct table. If you do not copy these tables, you will continue to see errors in the Transaction Server log where events cannot be delivered to all the subscribers, although this does not affect the delivery of the events to the correct queue.

4. Create new OCM mappings for all the tables listed in Steps 2 and 3 for the new Role (for example, role associated with the PD900 environment and the copied F90710 table and the role associated with the PD900 environment and the F90730 table) to point to Business Data - PROD. You must make mappings for both SYSTEM and Server Map.
5. On a JD Edwards EnterpriseOne client machine, run the program to create the table trigger on the new F90710 table. From a command prompt, navigate to this directory:

```
e900\system\bin32\
```

6. From the above directory, run this command:

```
dbtemplates.exe -create
```

When prompted ensure you log in as the new user (assigned to your new role) so that the program uses the new OCM mappings to locate the correct table in which to create the new trigger.

Henceforth when logging in as a user defined by this role, a generated Real Time Event sends the message into the appropriate F90710 table and updates the F90730 table.

7. Install a new Transaction server that will be used to process the events for each new Role/User.
8. Configure the newly installed Transaction Server with one of the users in the new Role as the bootstrap user.
9. Process a transaction that triggers an event for the new Role.
10. Start the newly installed Transaction Server and check the queue to make sure the event was delivered to the new Transaction Server.
11. Verify the previous configuration still functions as expected by logging in as any user *not* belonging to the new Role and generate an Event and see that the message goes into the F90710 (System - 900) and is delivered to the original Transaction Server.

Working with JMS Topic or JMS Queue

Based on your desired subscription model, you must choose to implement either JMS Topic or JMS Queue. You should examine these use case descriptions described in this appendix to determine whether you should use JMSTOPIC or JMSQUEUE subscription in your implementation of the Transaction Server:

- [Section C.1, "Use Case for JMS Topic"](#)
- [Section C.2, "Use Case for JMS Queue"](#)

C.1 Use Case for JMS Topic

The use case for JMS Topic includes these considerations:

- A distribution mechanism for publishing messages that are delivered to multiple subscribers.
- Multiple consumers require delivery of the message.
- There is a timing dependency between publishers and subscribers. The publisher must create a subscription in order for clients to be able to subscribe. The subscriber must remain continuously active to receive messages, unless it has established a durable subscription. In that case, messages published while the subscriber is not connected will be redistributed whenever it reconnects.

C.2 Use Case for JMS Queue

The use case for JMS Queue includes these considerations:

- A staging area that contains messages that have been sent and are waiting to be read. As the name queue suggests, the messages are delivered in the order sent. A message is removed from the queue once it has been read.
- Point-to-point solution based on Publisher/ Subscriber model.
- Only one consumer will get the message.
- The producer does not have to be running at the time the consumer consumes the message, nor does the consumer need to be running at the time the message is sent,
- Every message successfully processed is acknowledged by the consumer.

Testing and Validating the JMSQUEUE Configuration

If you have followed the steps to activate RTABOUT, a Real Time Event will be created whenever you Add, Change, or Delete an Address Book record. However, before performing any of those functions you should follow the procedure outlined in the following documents found in My Oracle Support. Access the document according to your Tools Release.

- E1: OUTBND: RTE Server (Transaction Server) Setup and Configuration for WebLogic - JMSQueue Subscriber (Doc ID 1137704.1)

Creating a JMS Topic in the JD Edwards EnterpriseOne Transaction Server Application

To create a JMS Topic in the Transaction Server Application:

1. Log on to the WebLogic Admin Console.
2. To create a JMS Topic for your JD Edwards EnterpriseOne, navigate to Home Page > Services > Messaging > JMS Modules, select your *JMS Module* and click the *New* button.
3. On Create a New JMS System Module Resource, select the radio button for *Topic*.

The process for creating the Connection Factory for this topic is the same as the JMS Connection Factory created for the queue. Ensure you use the same values as the ones used when setting up your JMSTOPIC Subscriber in P90702A.

Creating a Subscriber JMS Queue in WebLogic Server other than Transaction Server

Although it is supported it is not recommended that you create the subscriber queue in the Transaction Server container. This section describes the recommendation, which is to create the a subscriber JMS queue on a WebLogic Server other than the Transaction Server.

This appendix describes these tasks:

- [Appendix F.1, "Creating the JMS Server"](#)
- [Appendix F.2, "Creating the JMS Module"](#)
- [Appendix F.3, "Creating the JMS Resource"](#)

F.1 Creating the JMS Server

To create the JMS Server:

1. From the WebLogic Administration Console, select the domain in which you want to create the JMS Server.
2. Click on *YourDomain* > Services > Messaging > JMS Servers
3. In the right panel click the **New** button.

Create a New JMS Server


Back Next Finish Cancel

JMS Server Properties

The following properties will be used to identify your new JMS Server.

* Indicates required fields

What would you like to name your new JMS Server?

 * **Name:**

Specify persistent store for the new JMS Server.

Persistent Store:

Back Next Finish Cancel

4. On Create a New JMS Server, enter a value for the name of the JMS server to be created. In this guide, the name is **Test-JMSServer**.
5. Use the pulldown to select the Persistence Store.

Note: If Persistence Store does not exist you must create a new Persistence Store by clicking the **Create a New Store** button.

6. Click the **Next** button.

Create a New JMS Server

Back Next Finish Cancel

Select targets

Select the server instance or migratable target on which you would like to deploy this JMS Server.

Target:

Back Next Finish Cancel

7. On Create a New JMS Server, select the Weblogic server node and click the **Finish** button.

As shown below, at this point using the example in this guide a JMS server called **Test-JMSServer** is created. This **Test-JMSServer** is targeted to **AdminServer**. However this can be targeted to any other container available, which may not necessarily be AdminServer

Summary of JMS Servers

JMS servers act as management containers for the queues and topics in JMS modules that are targeted to them.
This page summarizes the JMS servers that have been created in the current WebLogic Server domain.

[Customize this table](#)

JMS Servers (Filtered - More Columns Exist)

<input type="checkbox"/>	Name	Persistent Store	Target	Current Target	Health
<input type="checkbox"/>	Test-JMSServer	Test-FileStore	AdminServer	AdminServer	OK

Showing 1 to 1 of 1 Previous | Next

F.2 Creating the JMS Module

To create a subscriber queue on WebLogic Server:

1. In the WebLogic Administration Console, select the domain into which you want to create the JMS Module.
2. Click on *YourDomain* > Services > Messaging > JMS Modules
3. In the right panel click the **New** button.

Create JMS System Module

[Back](#) [Next](#) [Finish](#) [Cancel](#)

The following properties will be used to identify your new module.

JMS system resources are configured and stored as modules similar to standard J2EE modules. Such resources include queues, topics, connection factories, templates, destination keys, quota, distributed queues, distributed topics, foreign servers, and JMS store-and-forward (SAF) parameters. You can administratively configure and manage JMS system modules as global system resources.

* Indicates required fields

What would you like to name your System Module?

* Name:

What would you like to name the descriptor file name? If you do not provide a name, a default will be assigned.

Descriptor File Name:

Where would you like to place the descriptor for this System Module, relative to the jms configuration sub-directory of your domain?

Location In Domain:

[Back](#) [Next](#) [Finish](#) [Cancel](#)

4. On Create JMS System Module, enter a value for the name of the JMS module to be created. In this guide, the name is **Test-SystemModule**.
5. Click the **Next** button.

Create JMS System Module

Back Next Finish Cancel

The following properties will be used to target your new JMS system module.

Use this page to select the server or cluster on which you would like to deploy this JMS system module. You can reconfigure targets later if you wish.

Targets :

Servers
<input checked="" type="checkbox"/> AdminServer
<input type="checkbox"/> S1

Back Next Finish Cancel

- On Create JMS System Module, select the target of Weblogic server node.
- Click the **Finish** button.

As shown below at this point, using the example in this guide, the **Test-SystemModule** is created. This **Test- SystemModule** is targeted to AdminServer. However this can be targeted to any other container .

Summary of JMS Modules

JMS system resources are configured and stored as modules similar to standard J2EE modules. Such resources include queues, topics, connection factories, templates, destination keys, quota, distributed queues, distributed topics, foreign servers, and JMS store-and-forward (SAF) parameters. You can administratively configure and manage JMS system modules as global system resources.

This page summarizes the JMS system modules that have been created for this domain.

[Customize this table](#)

JMS Modules	
Name	Type
Test-SystemModule	System

Showing 1 to 1 of 1 Previous Next

F.3 Creating the JMS Resource

Use this procedure to create a JMS resource such as connection factory, queue, or topic in the JMS Module you created in the previous section.

To create a JMS Resource:

- Click the JMS Module where you want to create a resource (in this guide **Test-SystemModule**).

Settings for Test-SystemModule

Configuration Subdeployments Targets Security Notes

This page displays general information about a JMS system module and its resources. It also allows you to configure new resources and access existing resources.

Name: Test-SystemModule The name of this JMS system module. [More Info...](#)

Descriptor File Name: jms/test-systemmodule-jms.xml The name of the JMS module descriptor file. [More Info...](#)

This page summarizes the JMS resources that have been created for this JMS system module, including queue and topic destinations, connection factories, JMS templates, destination sort keys, destination quota, distributed destinations, foreign servers, and store-and-forward parameters.

[Customize this table](#)

Summary of Resources

[New](#) [Delete](#) Showing 0 to 0 of 0 [Previous](#)

<input type="checkbox"/>	Name	Type	JNDI Name	Subdeployment	Targets
There are no items to display					

[New](#) [Delete](#) Showing 0 to 0 of 0 [Previous](#)

- On Settings for <Module_Name> , click the **New** Button.

Create a New JMS System Module Resource

[Back](#) [Next](#) [Finish](#) [Cancel](#)

Choose the type of resource you want to create.

Use these pages to create resources in a JMS system module, such as queues, topics, templates, and connection factories.

Depending on the type of resource you select, you are prompted to enter basic information for creating the resource. For targetable resources, like stand-alone queues and topics, connection factories, distributed queues and topics, foreign servers, and JMS SAP destinations, you can also proceed to targeting pages for selecting appropriate server targets. You can also associate targetable resources with subdeployments, which is an advanced mechanism for grouping JMS module resources and the members to server resources.

☒ **Connection Factory** Defines a set of connection configuration parameters that are used to create connections for JMS clients. [More Info...](#)

☐ **Queue** Defines a point-to-point destination type, which are used for asynchronous peer communications. A message delivered to a queue is distributed to only one consumer. [More Info...](#)

☐ **Topic** Defines a publish/subscribe destination type, which are used for asynchronous peer communications. A message delivered to a topic is distributed to all topic consumers. [More Info...](#)

☐ **Distributed Queue** Defines a set of queues that are distributed on multiple JMS servers, but which are accessible as a single, logical queue to JMS clients. [More Info...](#)

☐ **Distributed Topic** Defines a set of topics that are distributed on multiple JMS servers, but which are accessible as a single, logical topic to JMS clients. [More Info...](#)

- On Create a New JMS System Module Resource, select the radio button for the resource type to be created. In this example, **Connection Factory** is selected.

Create a New JMS System Module Resource

Back Next Finish Cancel

Connection Factory Properties

The following properties will be used to identify your new connection factory. The current module is Test-SystemModule.

* Indicates required fields

What would you like to name your new connection factory?

* Name: TestConnectionFactory

What JNDI Name would you like to use to look up your new connection factory?

JNDI Name: jms/TestConnectionFactory

The Connection Factory Subscription Sharing Policy Subscribers can be used to control which subscribers can access new subscriptions. Should subscriptions created using this factory be sharable?

Subscription Sharing Policy: Exclusive

The Client ID Policy indicates whether more than one JMS connection can use the same Client ID. Oracle recommends setting the Client ID policy to Unrestricted if sharing durable subscribers. Subscriptions created with different Client ID policies are always treated as independent subscriptions. What Client ID Policy would you like to use?

Client ID Policy: Restricted

A connection factory can limit the number of messages that can be queued for an asynchronous session. Should this connection factory impose a limit?

Maximum Messages per Session: 10

Should this connection factory create sessions that are JTA aware, and create XA queues and XA topics?

☒ XA Connection Factory Enabled

Back Next Finish Cancel

4. On Create a New JMS System Module Resource, enter Connection Factory name and JNDI name for this Connection Factory.
5. Click the Next button.

Create a New JMS System Module Resource

Back Next Finish Cancel

The following properties will be used to target your new JMS system module resource

Use this page to select a subdeployment to assign this system module resource. A subdeployment is a mechanism by which JMS resources are grouped and targeted to a server instance, cluster, or SAF agent. If necessary, you can create a new subdeployment by clicking the **Create a New Subdeployment** button. You can also reconfigure subdeployment targets later by using the parent module's subdeployment management page.

Select the subdeployment you want to use. If you select (none), no targeting will occur.

Subdeployments: (none) Create a New Subdeployment

What targets do you want to assign to this subdeployment?

Targets :

Back Next Finish Cancel

6. On Create a New JMS System Module Resource, use the pulldown to select the Subdeployment. If no existing Subdeployment can be used click on **Create a New Subdeployment** button.

Create a New Subdeployment

OK Cancel

Subdeployment Properties

The following properties will be used to identify your new subdeployment.

Subdeployment Name: Test-Subdeployment

OK Cancel

7. On Create a New Subdeployment, enter a name for a new Subdeployment.
8. Click the OK button.

Create a New JMS System Module Resource

[Back](#) [Next](#) [Finish](#) [Cancel](#)

The following properties will be used to target your new JMS system module resource

Use this page to select a subdeployment to assign this system module resource. A subdeployment is a mechanism by which JMS resources are grouped and targeted to a server instance, cluster, or SAF agent. If necessary, you can create a new subdeployment by clicking the **Create a New Subdeployment** button. You can also reconfigure subdeployment targets later by using the parent module's subdeployment management page.

Select the subdeployment you want to use. If you select (none), no targeting will occur.

Subdeployments: Test-Subdeployment ▼ [Create a New Subdeployment](#)

What targets do you want to assign to this subdeployment?

Targets :

Servers
<input type="checkbox"/> AdminServer

JMS Servers
<input checked="" type="checkbox"/> Test-JMSServer

[Back](#) [Next](#) [Finish](#) [Cancel](#)

9. On Create a New JMS System Module Resource, click the checkbox next to the JMS Server for which you just created a resource.
10. Click the **Next** button.

Create a New JMS System Module Resource

[Back](#) [Next](#) [Finish](#) [Cancel](#)

Choose the type of resource you want to create.

Use these pages to create resources in a JMS system module, such as queues, topics, templates, and connection factories.

Depending on the type of resource you select, you are prompted to enter basic information for creating the resource. For targetable resources, like stand-alone queues and topics, connection factories, distributed queues and topics, foreign servers, and JMS SAF destinations, you can also proceed to targeting pages for selecting appropriate server targets. You can also associate targetable resources with subdeployments, which is an advanced mechanism for grouping JMS module resources and the members to server resources.

<input type="radio"/> Connection Factory	Defines a set of connection configuration parameters that are used to create connections for JMS clients. More Info...
<input checked="" type="radio"/> Queue	Defines a point-to-point destination type, which are used for asynchronous peer communications. A message delivered to a queue is distributed to only one consumer. More Info...
<input type="radio"/> Topic	Defines a publish/subscribe destination type, which are used for asynchronous peer communications. A message delivered to a topic is distributed to all topic consumers. More Info...
<input type="radio"/> Distributed Queue	Defines a set of queues that are distributed on multiple JMS servers, but which are accessible as a single, logical queue to JMS clients. More Info...
<input type="radio"/> Distributed Topic	Defines a set of topics that are distributed on multiple JMS servers, but which are accessible as a single, logical topic to JMS clients. More Info...

11. On Create a New JMS System Module Resource, click the radio button for **Queue**.
12. Click the **Next** button.

Create a New JMS System Module Resource

Back

Next

Finish

Cancel

JMS Destination Properties

The following properties will be used to identify your new Queue. The current module is Test-SystemModule.

* Indicates required fields

* Name:

TestQueue

JNDI Name:

jms/TestQueue

Template:

None

Back

Next

Finish

Cancel

13. On Create a New JMS System Module Resource, enter a I and *JNDI Name* for this Queue.

14. Click the **Next** button.

Create a New JMS System Module Resource

Back

Next

Finish

Cancel

The following properties will be used to target your new JMS system module resource

Use this page to select a subdeployment to assign this system module resource. A subdeployment is a mechanism by which JMS resources are grouped and targeted to a server instance, cluster, or SAF agent. If necessary, you can create a new subdeployment by clicking the **Create a New Subdeployment** button. You can also reconfigure subdeployment targets later by using the parent module's subdeployment management page.

Select the subdeployment you want to use. If you select (none), no targeting will occur.

Subdeployments:

Test-Subdeployment

Create a New Subdeployment

What targets do you want to assign to this subdeployment?

Targets :

JMS Servers

Test-JMSServer

Back

Next

Finish

Cancel

15. On Create a New JMS System Module Resource, select the Subdeployment.

16. Click the **Finish** button.

17. As shown below, verify that both the xxxx

Settings for Test-SystemModule

Configuration

Subdeployments

Targets

Security

Notes

This page displays general information about a JMS system module and its resources. It also allows you to configure new resources and access existing resources.

Name:

Test-SystemModule

The name of this JMS system module. [More Info...](#)

Descriptor File Name:

Test-SystemModule-jms.xml

The name of the JMS module descriptor file. [More Info...](#)

This page summarizes the JMS resources that have been created for this JMS system module, including queue and topic destinations, connection factories, JMS templates, destination sort keys, destination quota, distributed destinations, foreign servers, and store-and-forward parameters.

Customize this table

Summary of Resources

New

Delete

Showing 1 to 2 of 2

Previous

Next

<input type="checkbox"/>	Name	Type	JNDI Name	Subdeployment	Targets
<input type="checkbox"/>	TestConnectionFactory	Connection Factory	TestConnectionFactory	Test-Subdeployment	Test-JMSServer
<input type="checkbox"/>	TestQueue	Queue	TestQueue	Test-Subdeployment	Test-JMSServer

New

Delete

Showing 1 to 2 of 2

Previous

Next

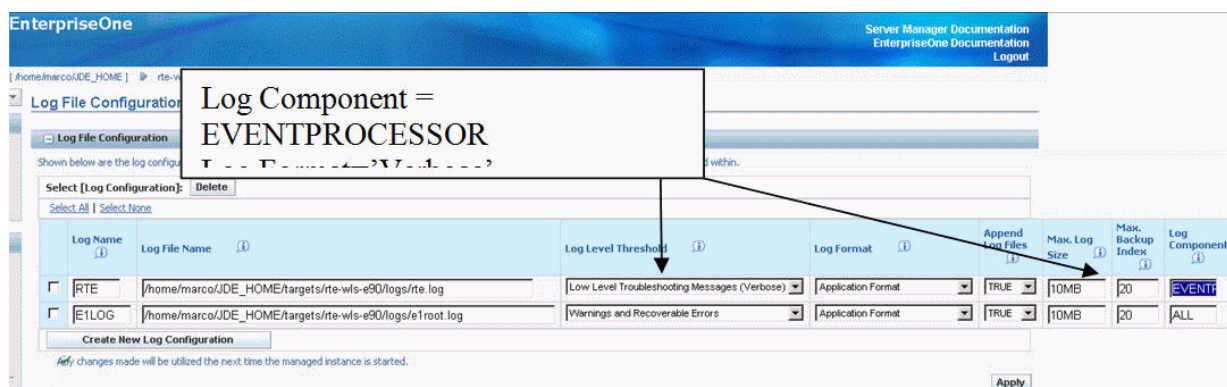
F-8 Transaction Server Reference Guide

Creating an Oracle JD Edwards EnterpriseOne Service Request for Transaction Server Support

To minimize the resolution time for any Service Requests you might have to create with Global Support, it is recommended that you provide this information:

- JD Edwards EnterpriseOne Enterprise Server JDE.INI file
- Exact JD Edwards EnterpriseOne Tools Release on which you are running (for example, 9.1.1.1)
- Indicate that you are running Transaction Server on Oracle WebLogic Server and the exact release and version your application server (for example, WebLogic Server 10.3.5)
- The content of the `config` folder within the Transaction Server Agent where the Transaction Server instance resides
- Logs from Server Manager generated when starting the Transaction Server

Tip: To obtain a more detailed set of logs, you can use Server Manager to enable enhanced logging. For the *Log Name* with a value *RTE*, change the *Log Components* value to *EVENTPROCESSOR* as shown below.



- Once the additional `jdelog.properties` definition is added as described in the preceding step, stop the Transaction Server, clear the RTE logs and produce the logs generated after going through the steps outlined in [Appendix D, "Testing and Validating the JMSQUEUE Configuration"](#).

-
- Description of the results obtained when running the test described in [Appendix D, "Testing and Validating the JMSQUEUE Configuration"](#)
Specifically, you should be able to provide the status of the F90710 record after the RTE is started.

Clustering for the JD Edwards Transaction Server on WebLogic Server

This appendix discusses these topics:

- [Section H.1, "Understanding WebLogic Clustering"](#)
- [Section H.2, "Creating a Cluster on WebLogic Server"](#)
- [Section H.3, "Creating the Servers"](#)
- [Section H.4, "Creating Machines"](#)
- [Section H.5, "Associate Machines with Servers"](#)
- [Section H.6, "Cluster the Servers"](#)
- [Section H.7, "Setup the Cluster for Load Balancing"](#)
- [Section H.8, "Setup the Cluster for Failover"](#)
- [Section H.9, "Installing the Transaction Server RTE Component on WebLogic Server using Server Manager"](#)
- [Section H.10, "Using JDBC Persistent Stores"](#)
- [Section H.11, "Create JDBC Datasources"](#)
- [Section H.12, "Create JDBC Persistent Stores"](#)
- [Section H.13, "Associate a JMS Server with a Persistent Store"](#)
- [Section H.14, "Additional Configuration for Using Horizontal Cluster"](#)

H.1 Understanding WebLogic Clustering

Clustering permits the deployment of application components and services to several machines with a single face to the client. When a client requests a service, it should make no difference if the service runs on a single server or across a number of servers. The clustering abstraction provides you with a clear route to improving the performance and scalability of the applications with increased administration of hardware and network resources. WebLogic clustering provides these benefits:

- Scalability

A clustered solution allows you to create additional capacity by introducing more servers to the cluster, thereby reducing the load on existing servers. For the JD Edwards EnterpriseOne Transaction Server with the Real Time (RTE) Application, you can add as many servers and messaging engines as you like to meet the

performance requirement. The RTE application on WebLogic achieves scalability with the implementation of a messaging engine policy.

- Load balancing

The ability to distribute requests across all members of the cluster, according to the workload on each server. WebLogic handles load balancing as defined by the Load Balancing Algorithm that you configure on cluster setup. The Transaction Server with RTE on WebLogic only supports round-robin load balancing policy for the RTE application.

- High availability

A mix of features that ensure applications and services are available even if a server or machine fails. Clients can continue to work with little or no disruption in a highly available environment. RTE on WebLogic achieves high availability with the implementation of a messaging engine policy.

H.2 Creating a Cluster on WebLogic Server

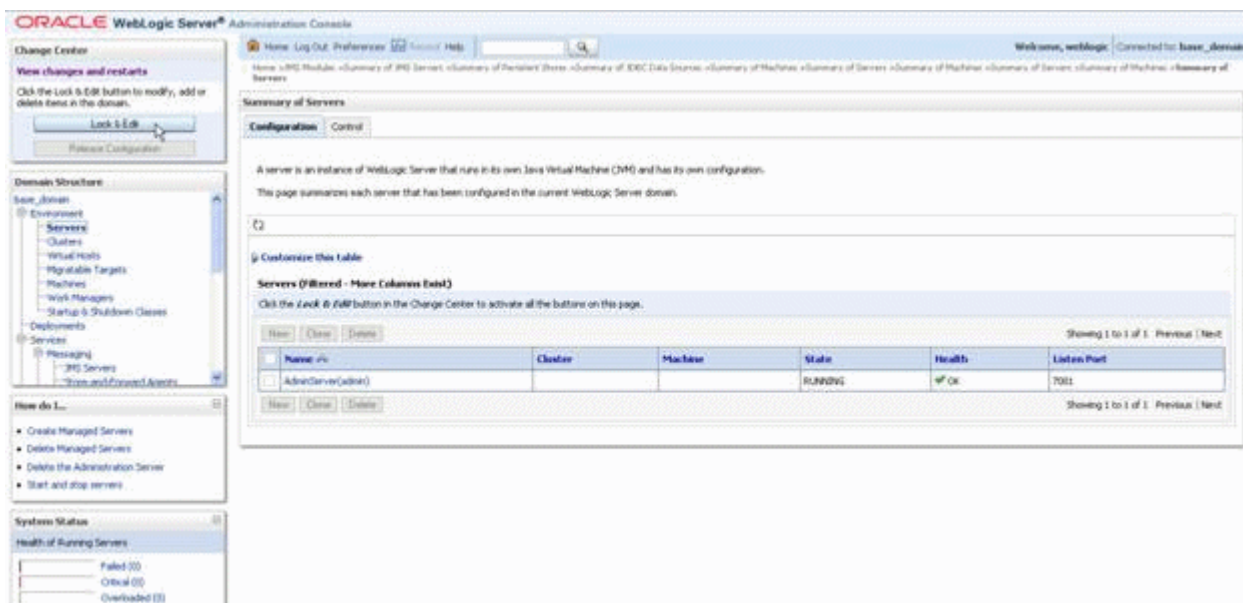
For advanced configuration of the Oracle WebLogic Server refer to this link:

http://download.oracle.com/docs/cd/E15523_01/web.1111/e13709/setup.htm

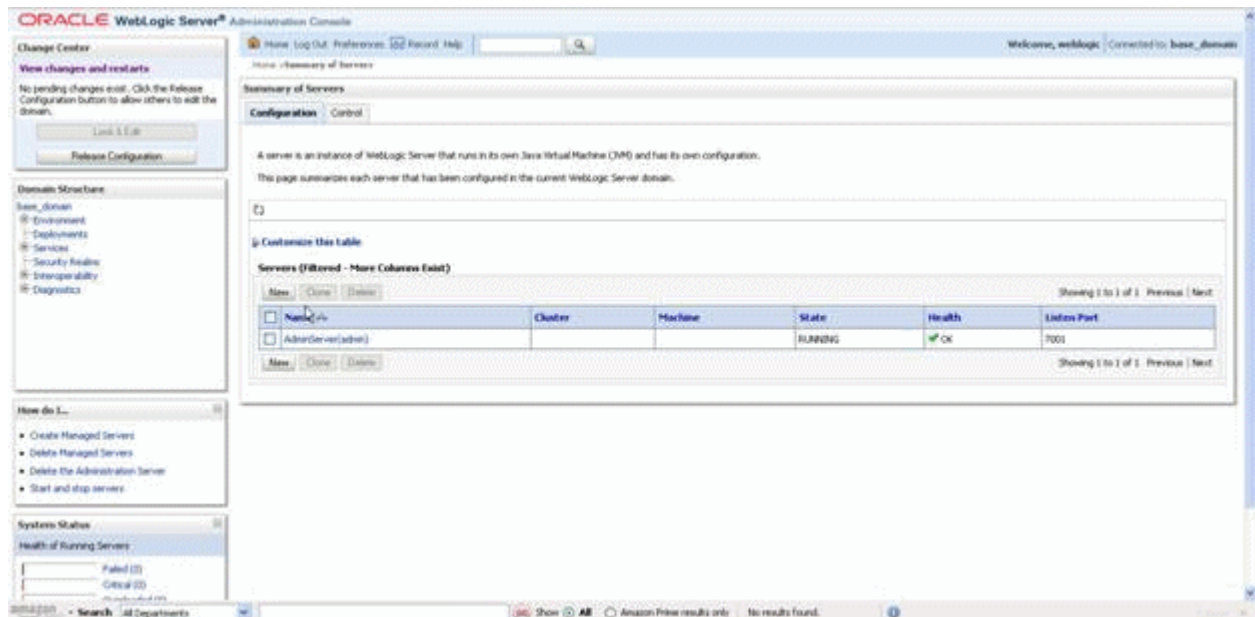
The section discusses these topics:

- Creating the Servers
- Creating Machines
- Associate Machines with Servers
- Cluster the Servers
- Setup the Cluster for Load Balancing
- Setup the Cluster for Failover

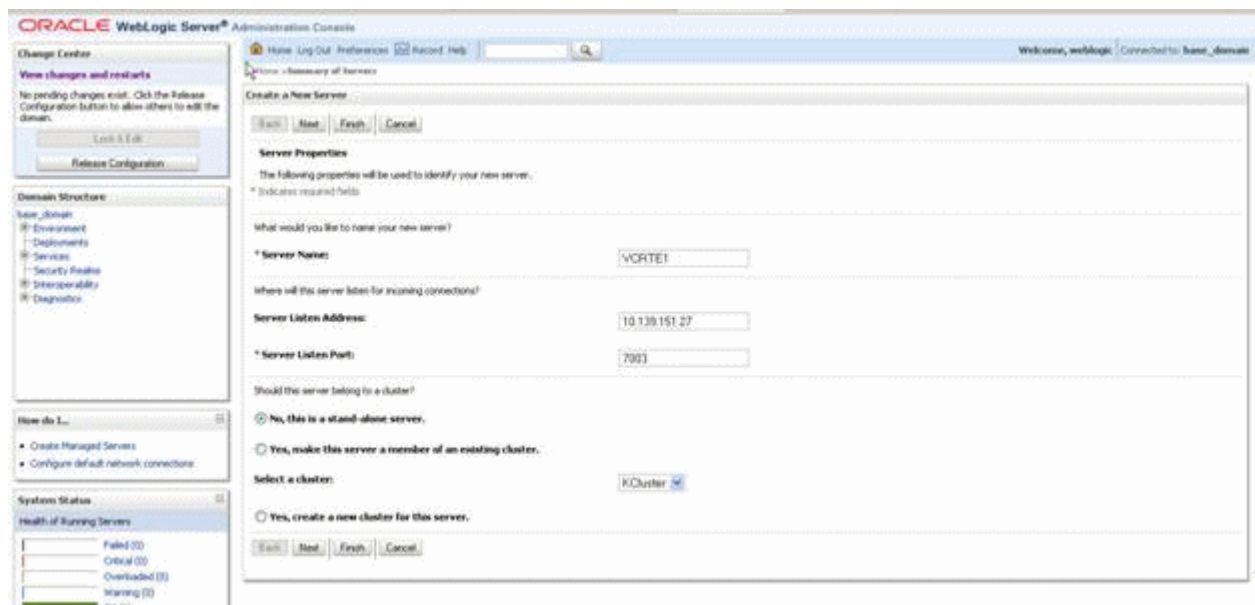
H.3 Creating the Servers



1. On Summary of Servers, click the **Lock and Edit** button at the bottom of the page.



2. On Summary of Servers, click the **New** Button.



3. On Create a New Server, complete these fields:

- *Server Name*

Enter the name of the new server.

- *Server Listen Address*

Enter the IP address of the physical machine on which this instance will run.

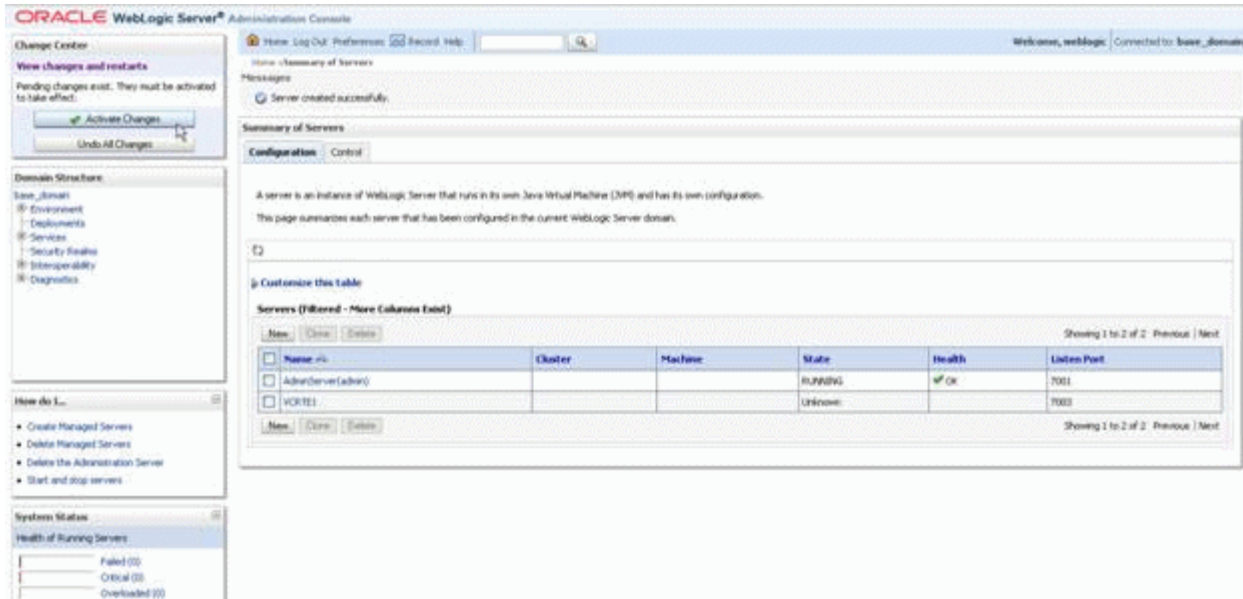
Note: If you want to setup a horizontal Cluster, you should create a server with the remote Server Listen Address.

■ *Server Listen Port*

Enter a valid unused port on which this server will listen.

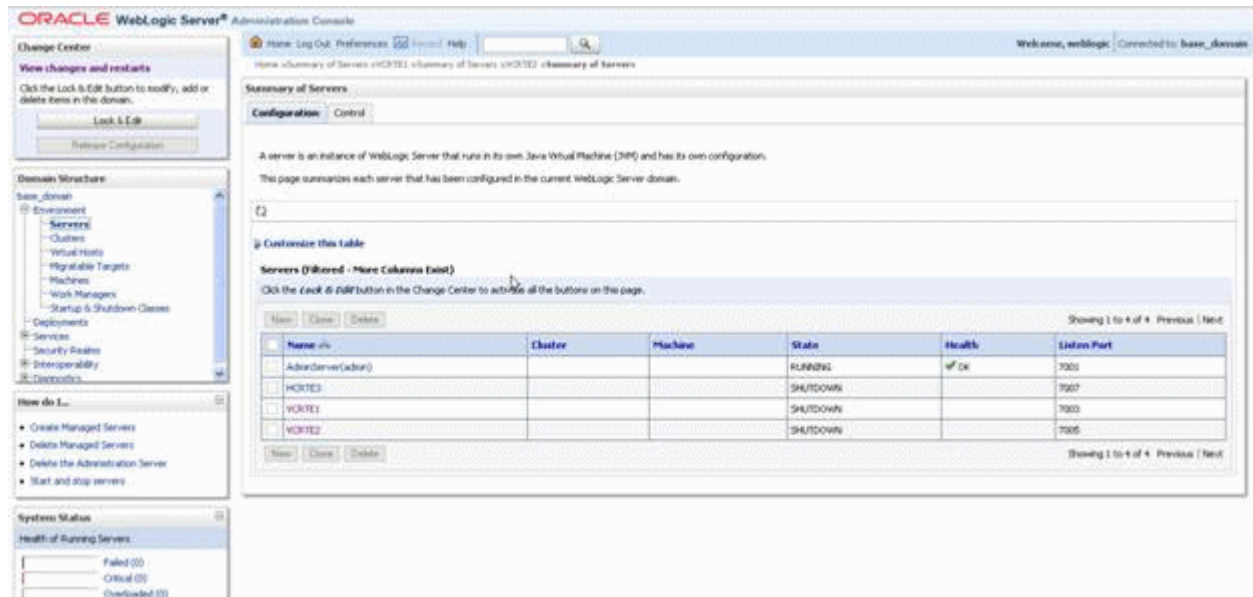
Note: If you want to setup a horizontal Cluster, you should create a server with a Remote Server Listen Port.

4. On Create a New Server, with the entered values accepted, click the Finish button.



5. On Create a New Server, click the **Activate Changes** button at the bottom of the screen.
6. Repeat the steps this procedure for any additional servers you want to make part of the cluster.
7. If you want to setup a horizontal Cluster, you should create a server with the remote Server listen Address and Remote Server Listen Port.

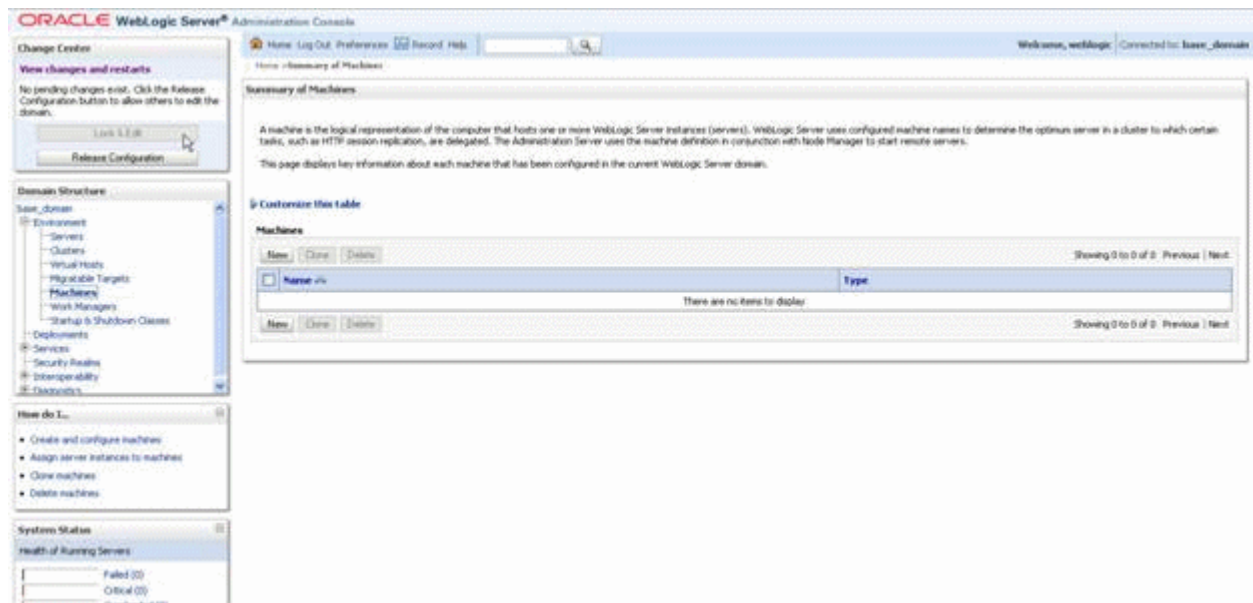
When you complete this procedure, your list of server will be similar to this:



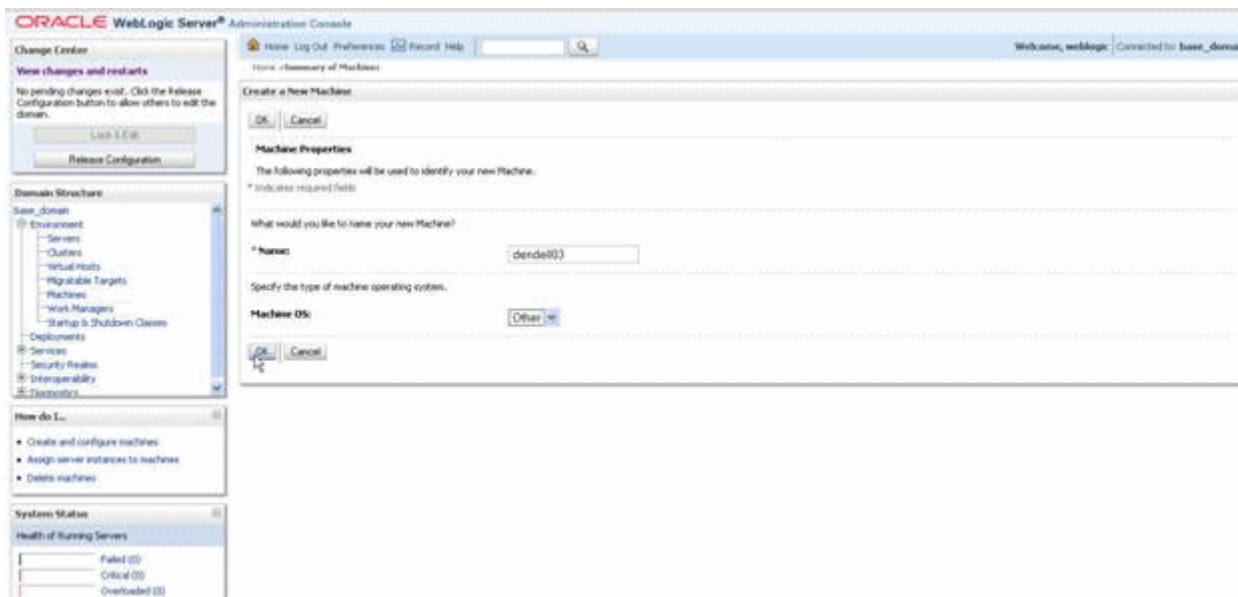
H.4 Creating Machines

Use this procedure to create the machines:

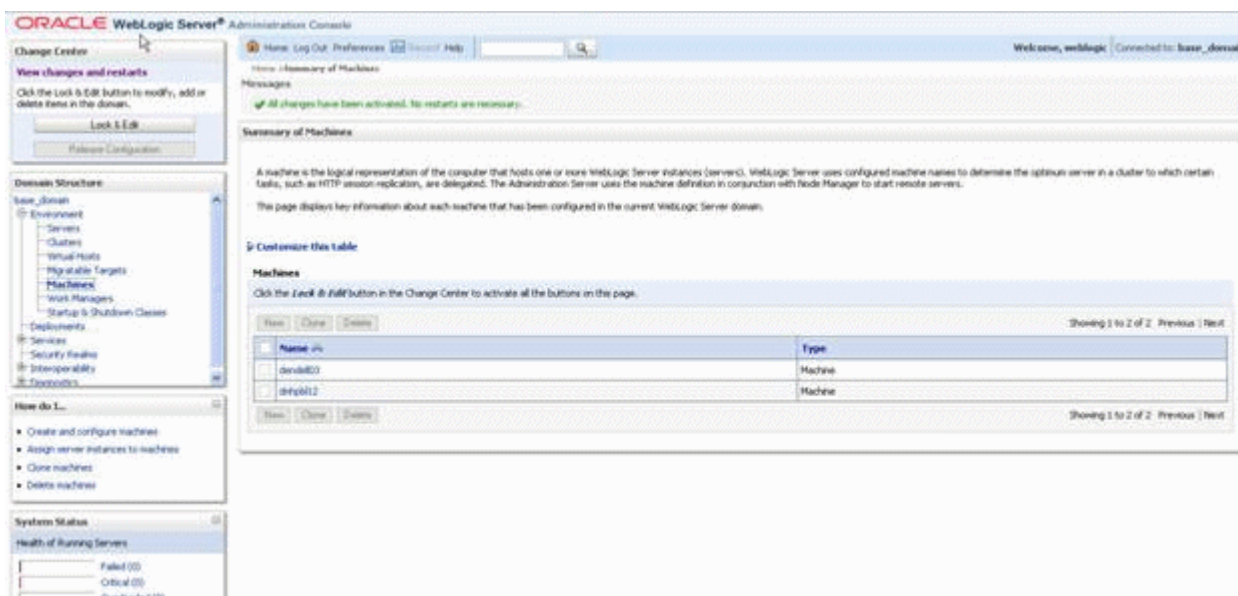
1. Click Environment > Machines > Lock & Edit.



2. On Summary of Machines, click the New button.



3. On Create a New Machine, complete these fields:
 - Name
Enter the machine name of the host for this clustered server.
 - Machine OS
Use the pulldown to select the operating system. Choose Other for all non-UNIX operating systems.
 4. On Create a New Machine, click the OK button.
 5. Click Activate Changes.
 6. Repeat the steps in this procedure if your cluster spans multiple machines or if multiple server instances will run on individual machines in the cluster.
- When you complete this procedure, your list of server will be similar to this:

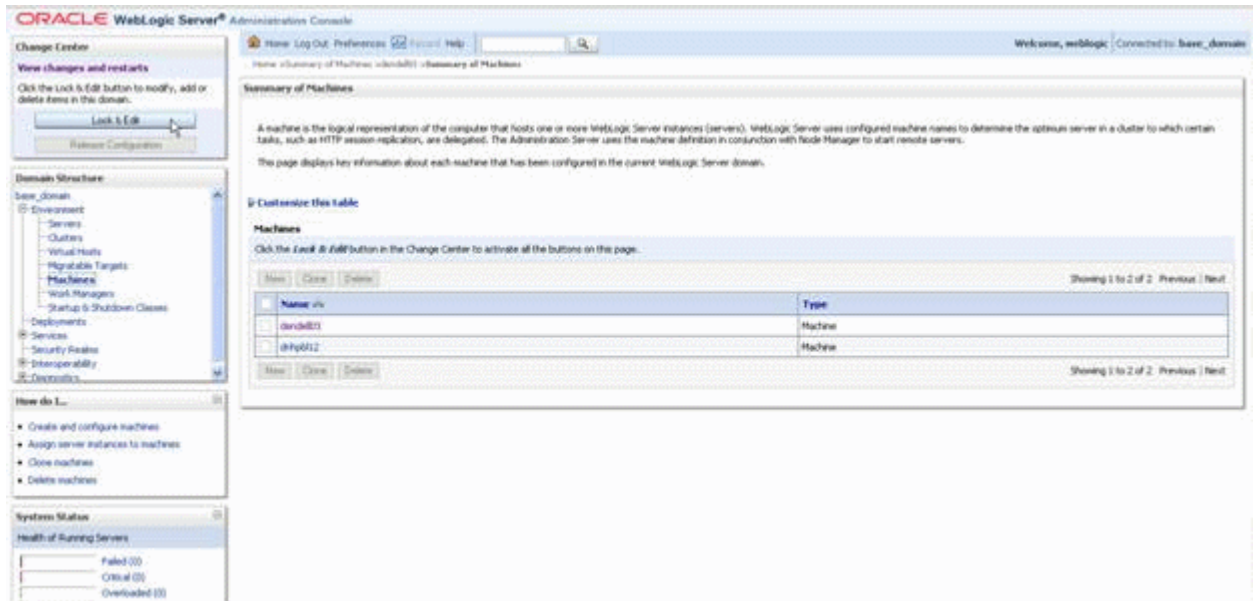


Note: For the Horizontal Node ensure that the listen address for the server and the machine are *exactly* the same. This should be the ip of the physical machine, not "localhost" or 127.0.0.1.

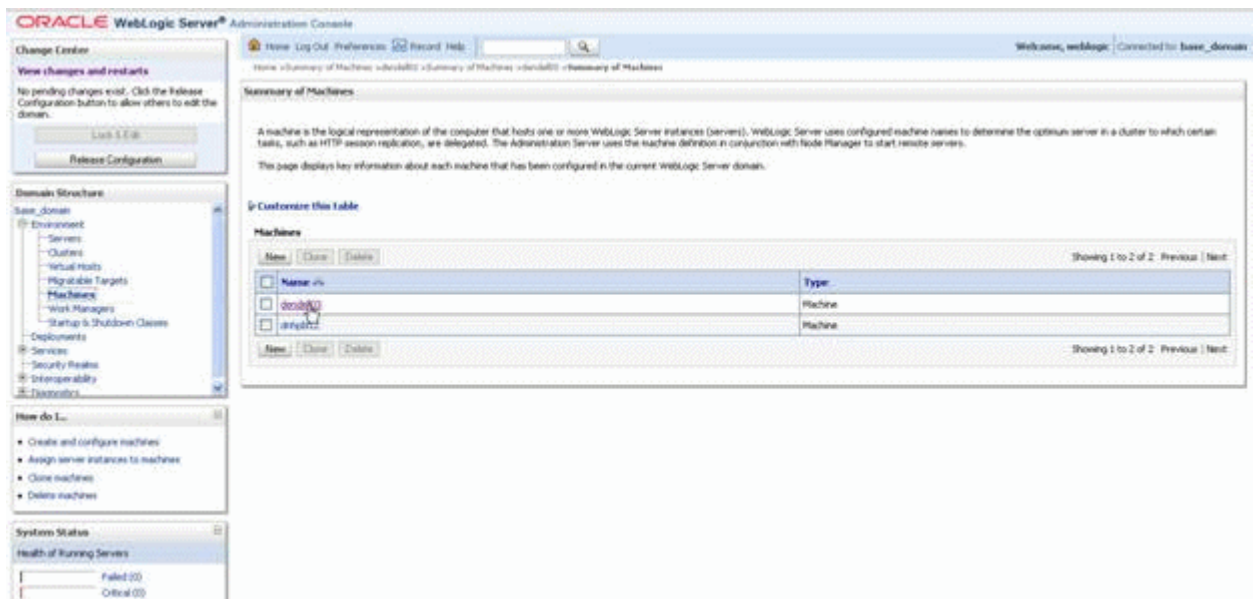
H.5 Associate Machines with Servers

Use this procedure to associate machines with Servers:

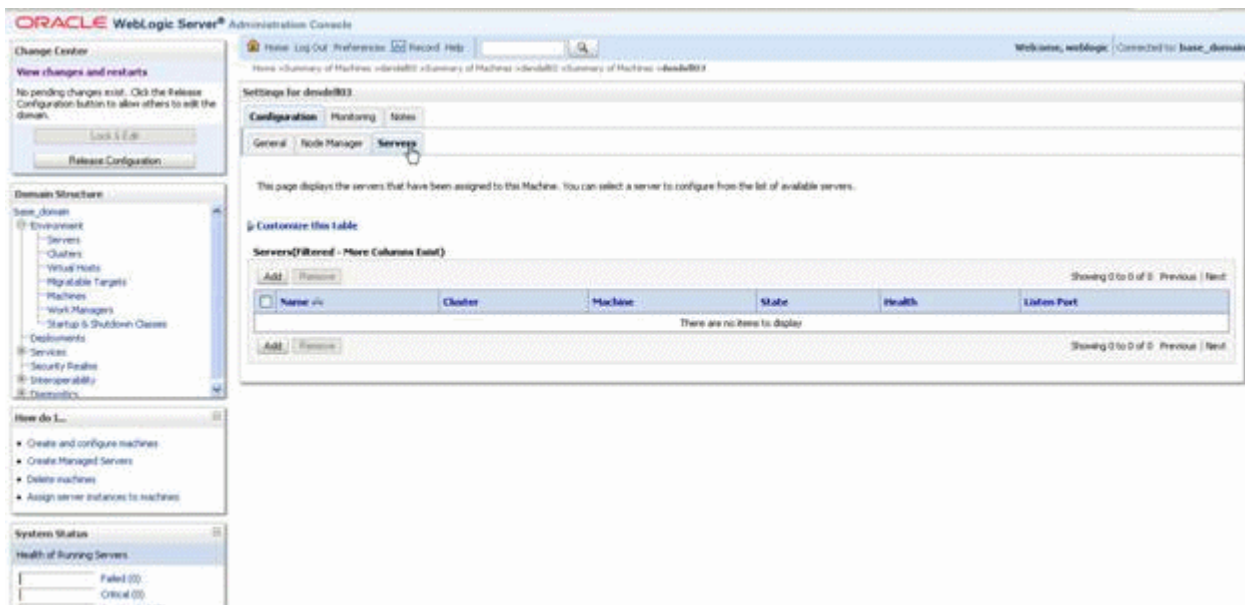
1. Go to Environment > Machines.



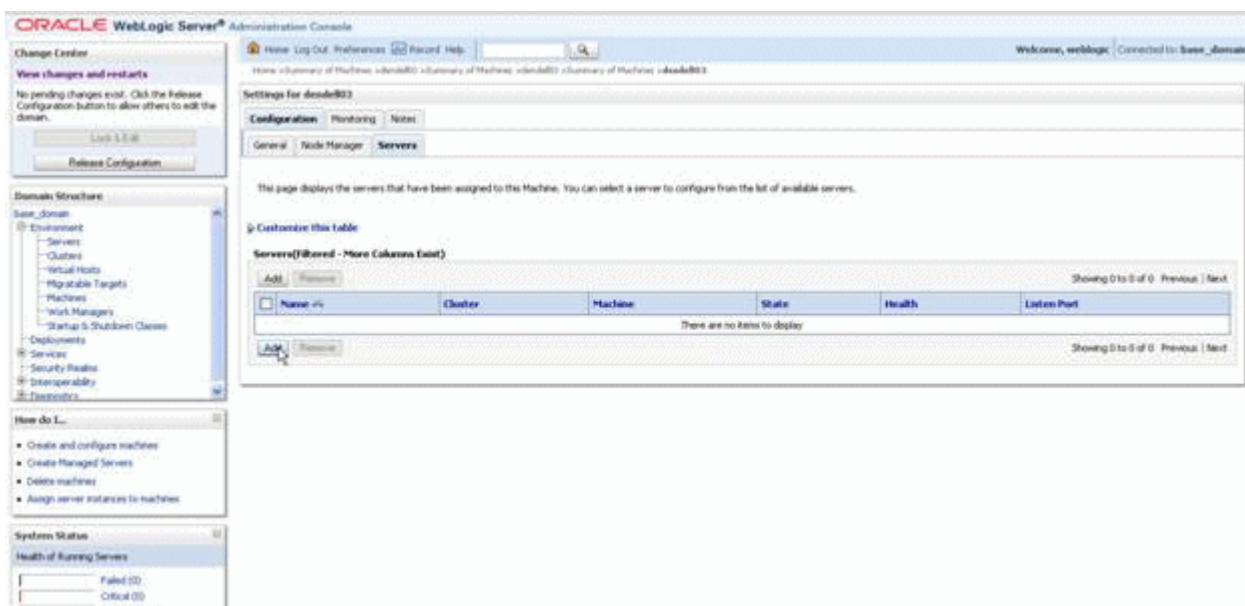
2. On Summary of Machines, click the Lock & Edit button at the bottom of the screen.



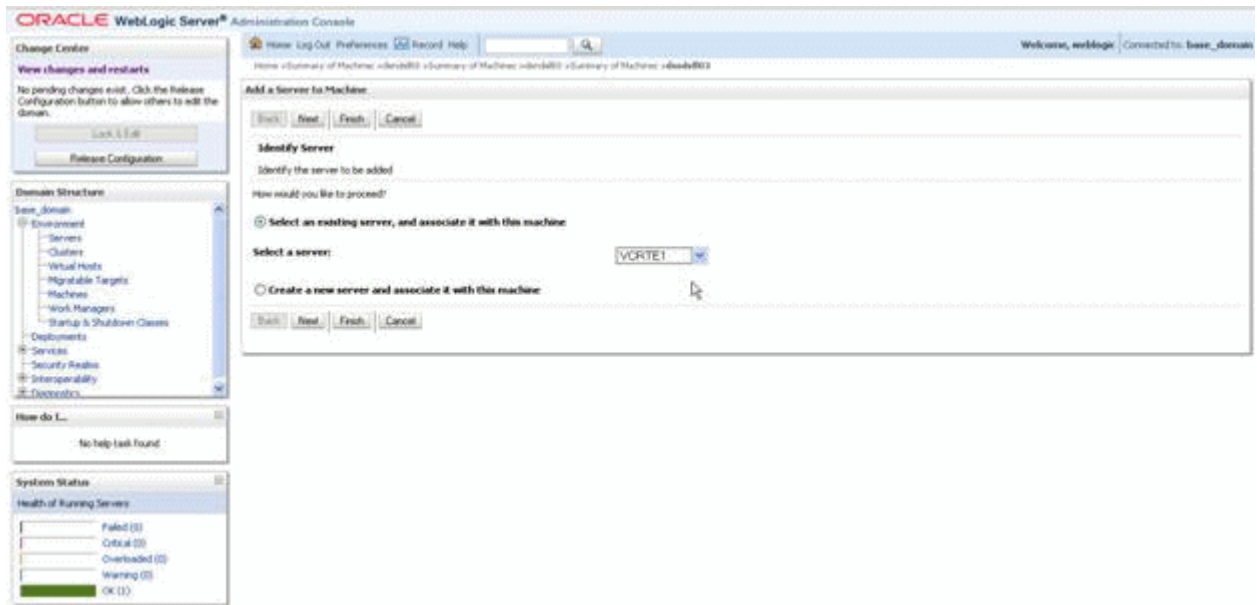
3. In the Machines list, select the machine for the vertical nodes. If setting up horizontal cluster, select the machine for the horizontal node.



4. In Settings for <machine>, click on the Servers tab.

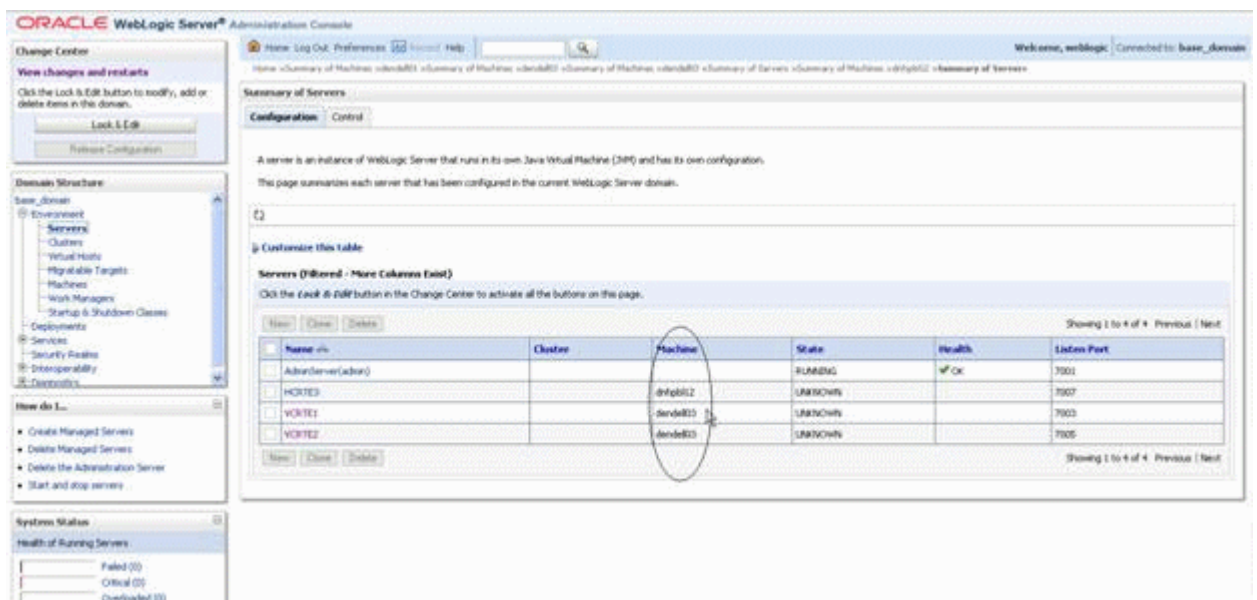


5. In Customize this table, click the Add button.



6. On Identify Server, complete these fields:
 - Check this radio button:
Select an existing server, and associate it with this machine
 - Select a server:
Use the pulldown to select an existing server.
7. Repeat the steps in this procedure if your cluster spans multiple machines or if multiple server instances will run on individual machines in the cluster.
8. Click the Finish button.
9. Click the Activate Changes button.

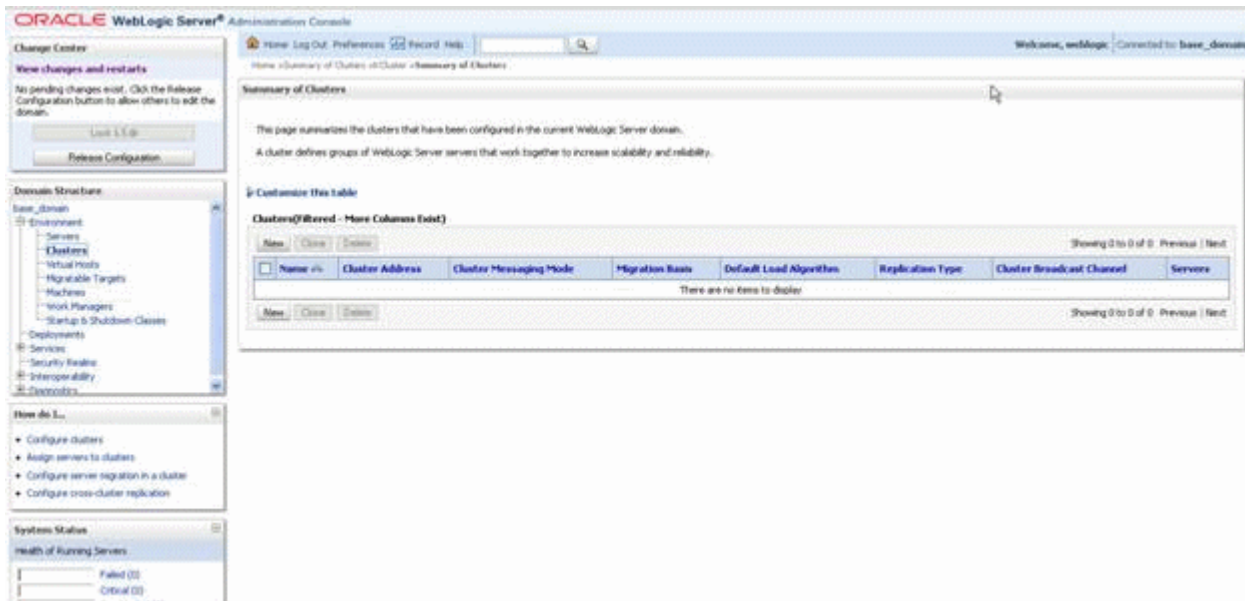
When you complete this procedure, your list of server will be similar to this:



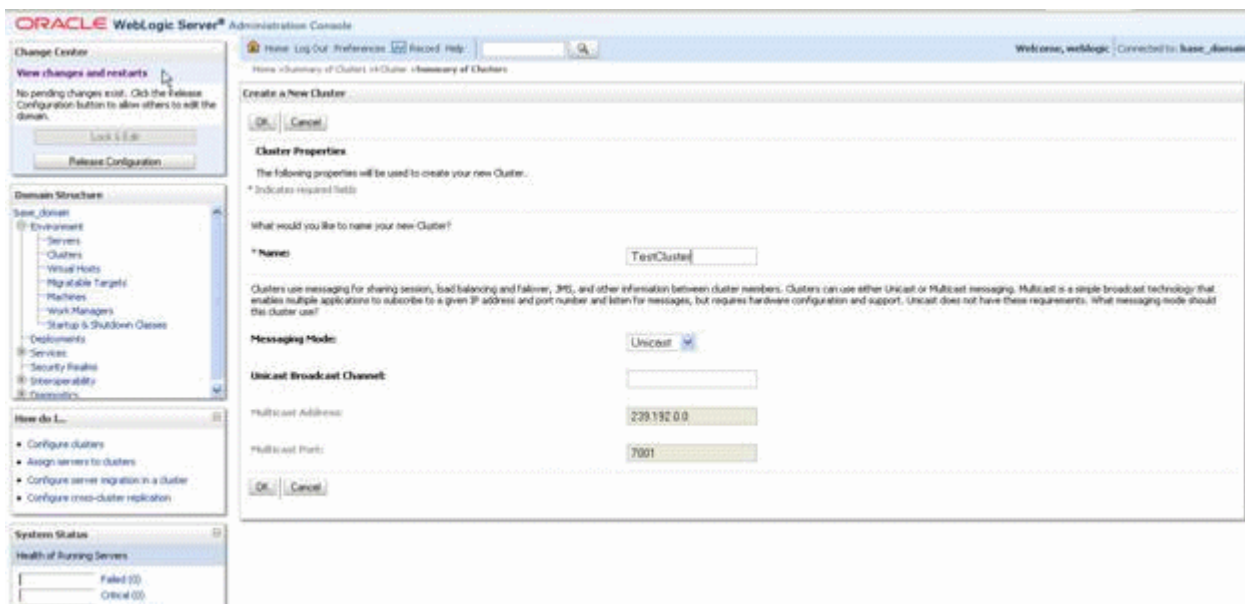
H.6 Cluster the Servers

Use this procedure to cluster the servers.

1. Go to Environment > Clusters.



2. On Summary of Clusters, click the Lock & Edit button at the bottom of the screen.
3. Click the New button.



4. On Cluster Properties, complete these fields:
 - *Name*
Enter a name for the cluster.
 - *Messaging Mode*
Use the pulldown to select Unicast.

- *Unicast Broadcast Channel*

Leave this field blank.

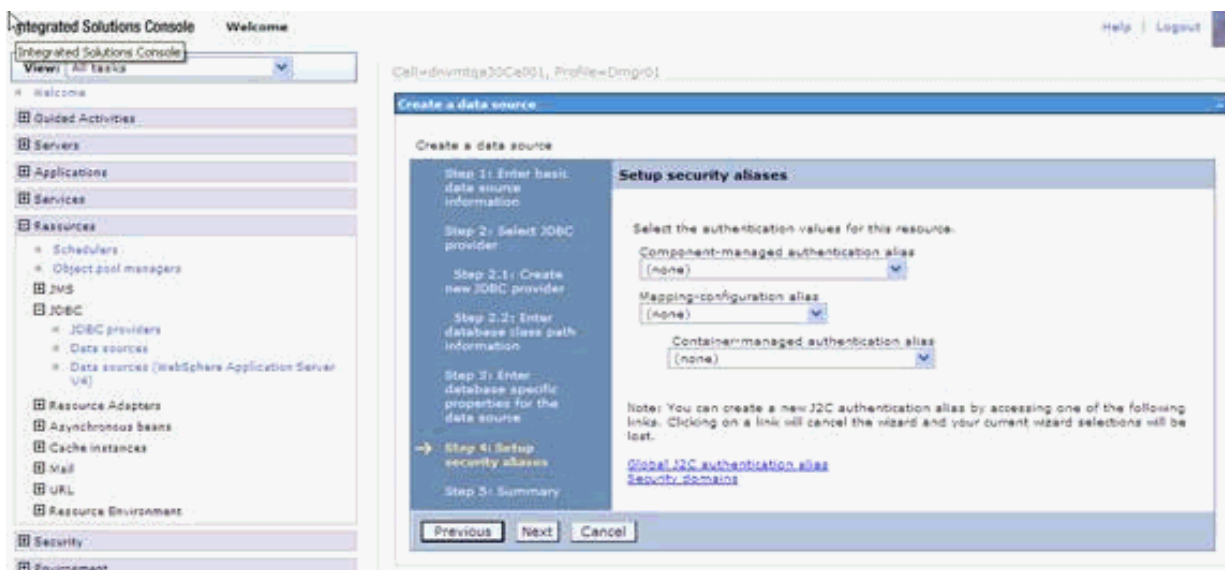
- *Multicast Address*

Accept the default value.

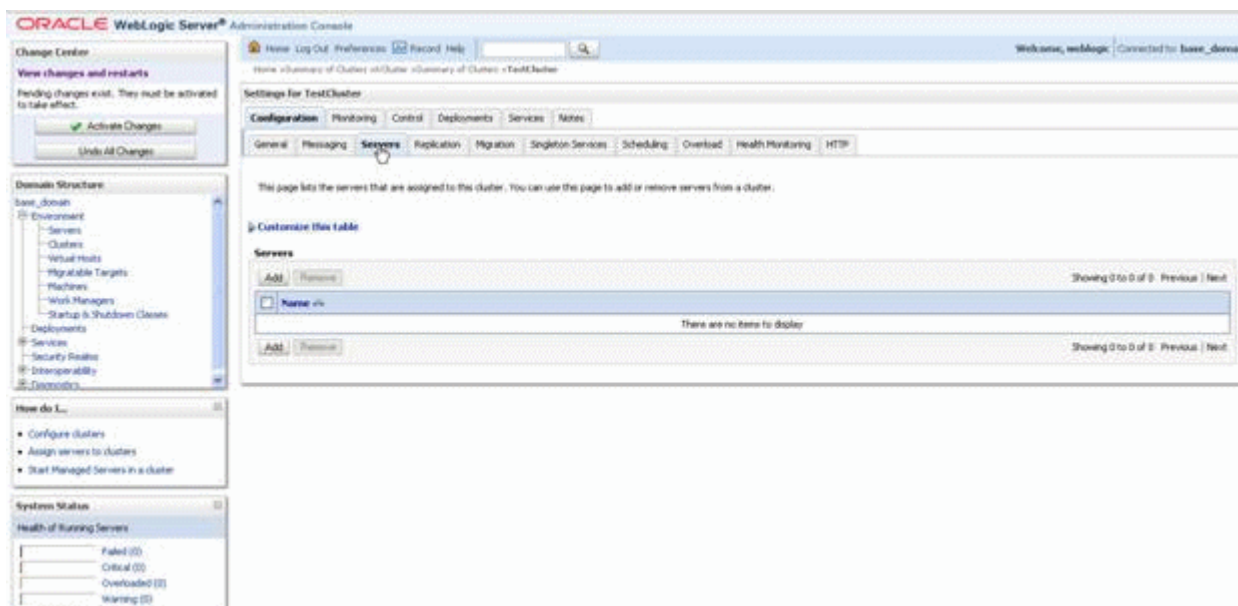
- *Multicast Port*

Accept the default value.

5. Click the OK button.



6. On Summary of Clusters, click on the newly-defined cluster name.



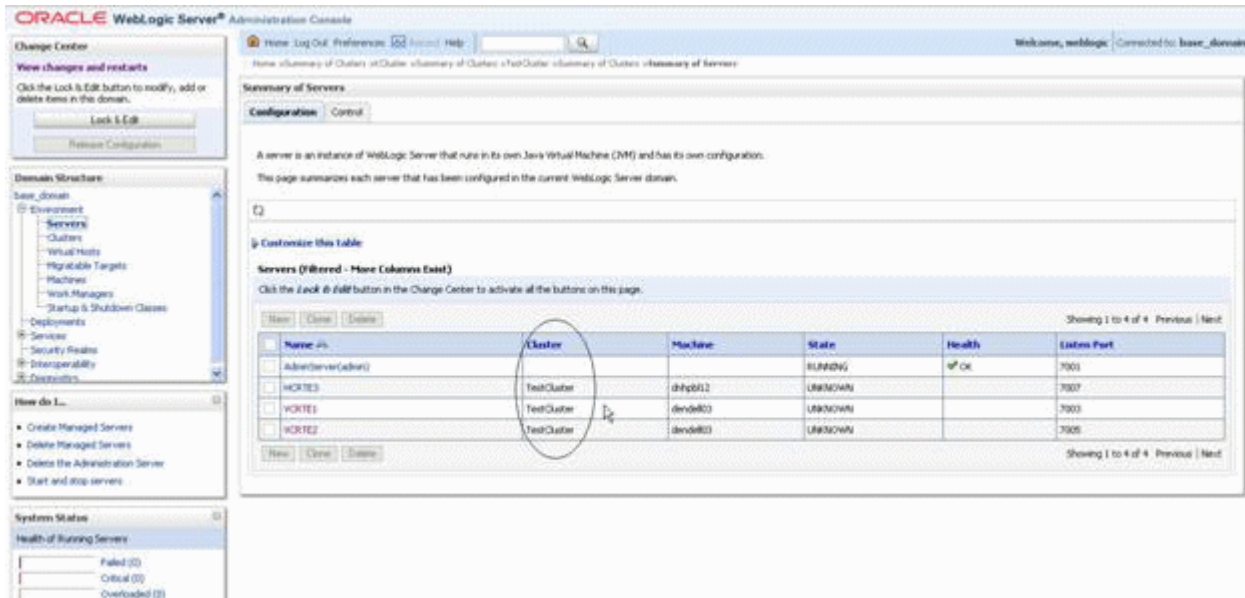
7. On Settings for <cluster>, click the **Servers** tab.

8. Click on the Add button.

9. Select the server.

10. Click next.
11. Repeat the preceding two steps for any additional servers.
12. Click the Activate Changes button.

When you complete this procedure, your list of server will be similar to this:

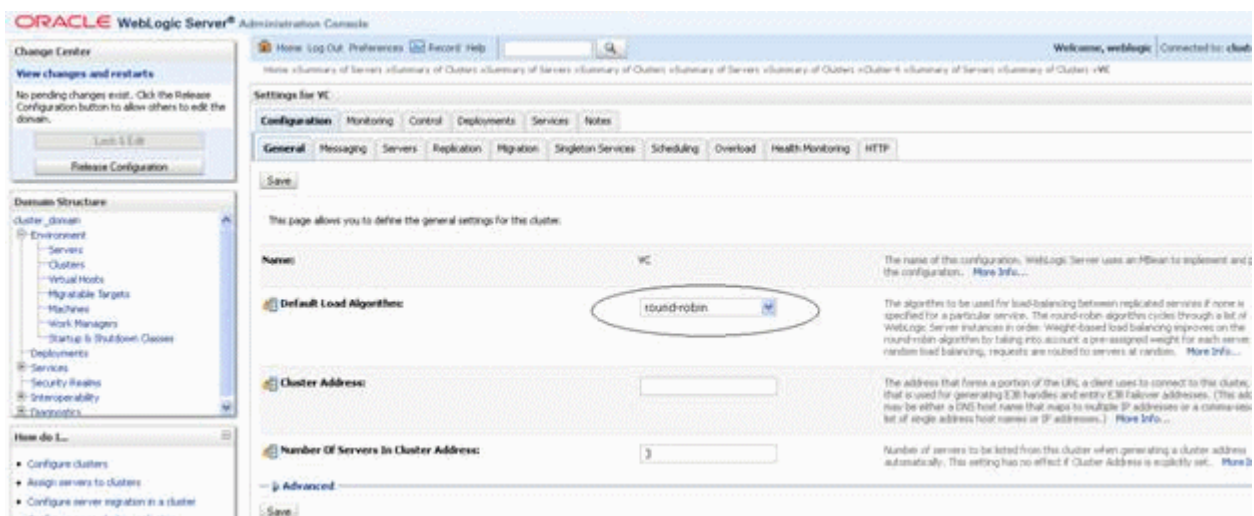


13. Restart the admin server and all node managers.

H.7 Setup the Cluster for Load Balancing

A cluster is setup with load balancing feature with round-robin algorithm by default. However the algorithm to be used can be changed based on requirement.

1. Go to click Environment > Clusters.
2. Click on the cluster name.
3. Go to General tab.



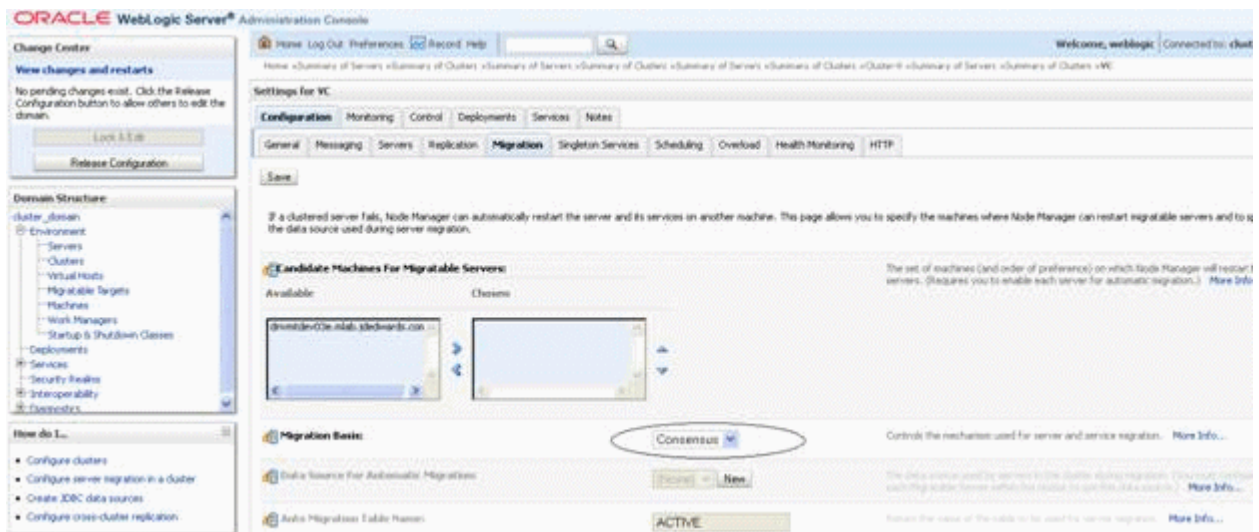
4. Click the Lock & Edit button.

5. Default Load Algorithm field contains several algorithms in the dropdown. Select the one you want to use. Different load balancing algorithms are round-robin, weight-based and random.
6. Click the Save button.
7. Click the Activate Changes button.

H.8 Setup the Cluster for Failover

A cluster is setup with load balancing feature with round-robin algorithm by default. However the algorithm to be used can be changed based on requirement.

1. Go to Environment > Clusters.
2. Click on the cluster name.
3. Go to Migration tab
4. Click the Lock & Edit button.
5. In the Candidate Machines For Migratable Servers field, move the machines Available to Choose.



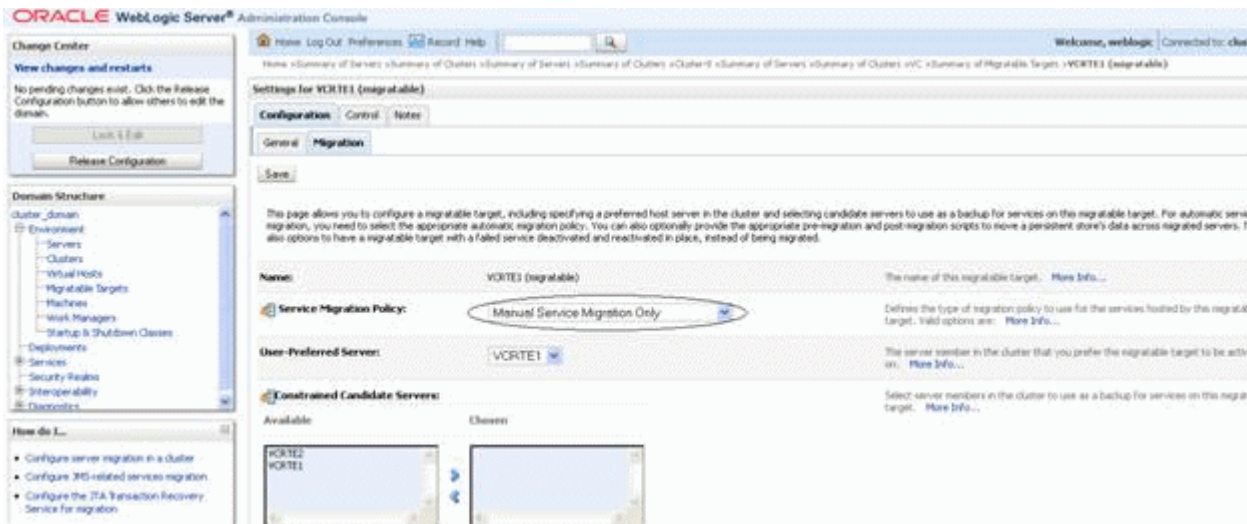
6. On Settings for VC, with the Configuration tab selected, complete this field:

- *Migration Basis*

Use the pulldown to select Consensus.

Note: You can also select database as Migration Basis with additional setup.

7. Click Environment, Migratable Targets.
8. Click on the migratable target.
9. Go to Migration tab.



10. On Settings for <machine> (migratable), with the Migration tab selected, complete this field:

- *Server Migration Policy*

Use the pulldown to select between these available policies:

- Manual Service Migration Only
- Auto-Migrate Exactly-Once Services
- Auto-Migrate Failure-Recovery Services

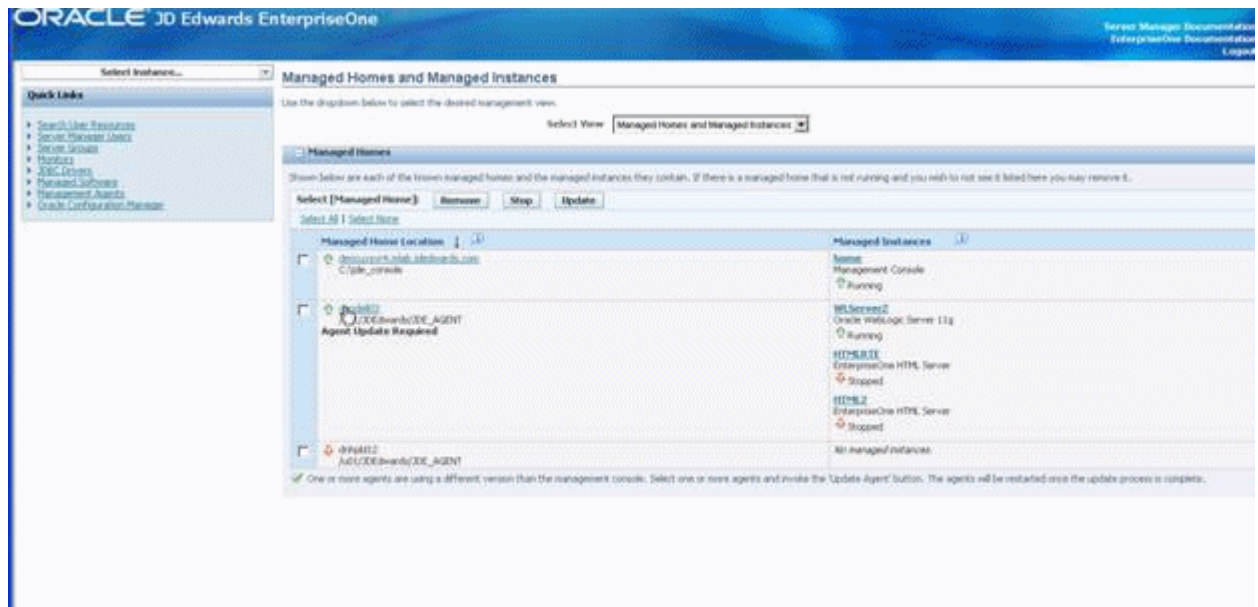
11. Click the Save button.

12. Click the Activate Changes button.

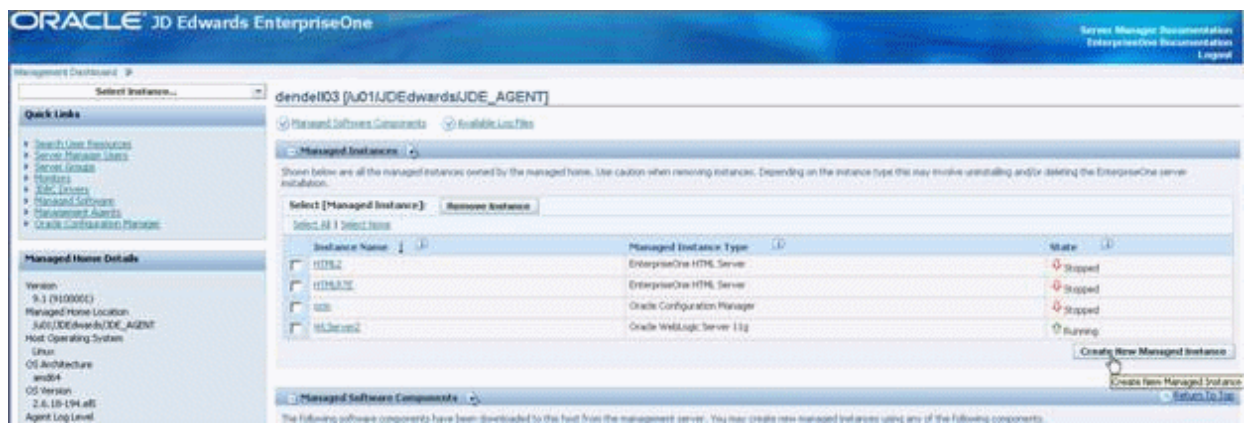
H.9 Installing the Transaction Server RTE Component on WebLogic Server using Server Manager

Use this procedure to install the RTE application to the cluster via Server Manager.

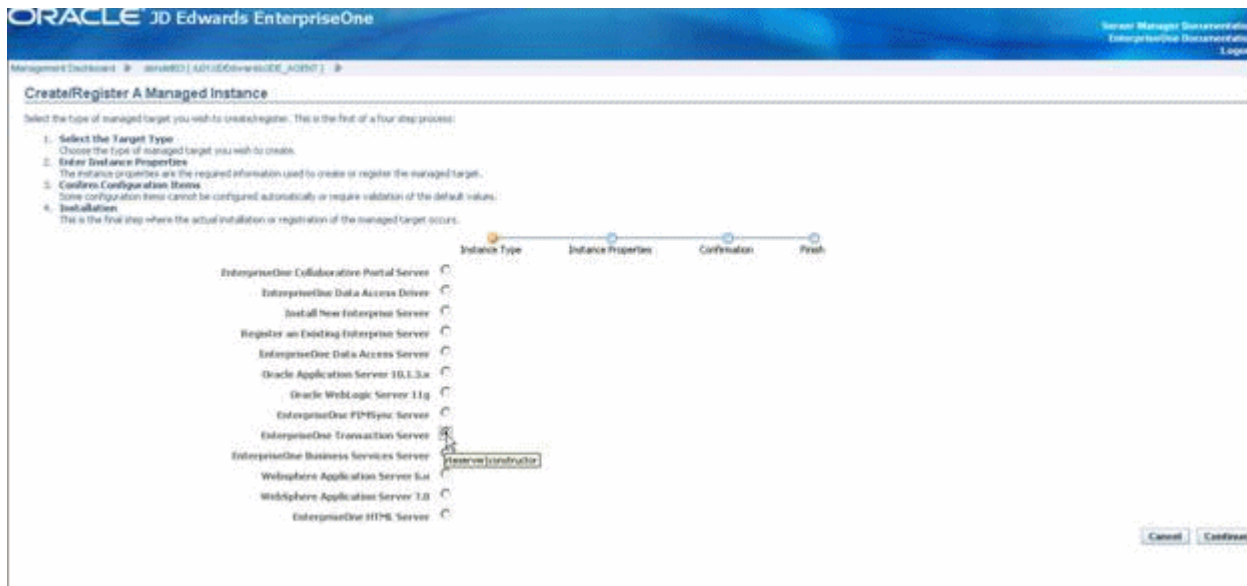
1. Install the JD Edwards EnterpriseOne product Server Manager on a machine with the Microsoft Windows operating system.
2. Install the Server Manager agent on the machine that is running the Oracle WebLogic Administration Console.
3. Register the WebLogic server through Server Manager.
4. Logon to the Server Manager console.



- On Managed Homes and Managed Instances, click on the machine running the registered Oracle WebLogic Server.



- Click the **Create New Managed Instance** button.



7. On Create/Register A Managed Instance, Instance Type, click this radio button:
EnterpriseOne Transaction Server
8. Click the Continue button.



9. On Create/Register A Managed Instance, Instance Properties, complete the fields:

- *Server Group*
Accept the default value.
- *Instance Name*
Enter an instance name.
- *J2EE Server/Cluster*
Use the pulldown to select the Oracle WebLogic cluster.
- *HTTP Port*
Accept the auto populated value.

Note: You should not manually change this value.

- *Software Component*

Use the pulldown to select a valid JD Edwards RTE application software component for your installation.

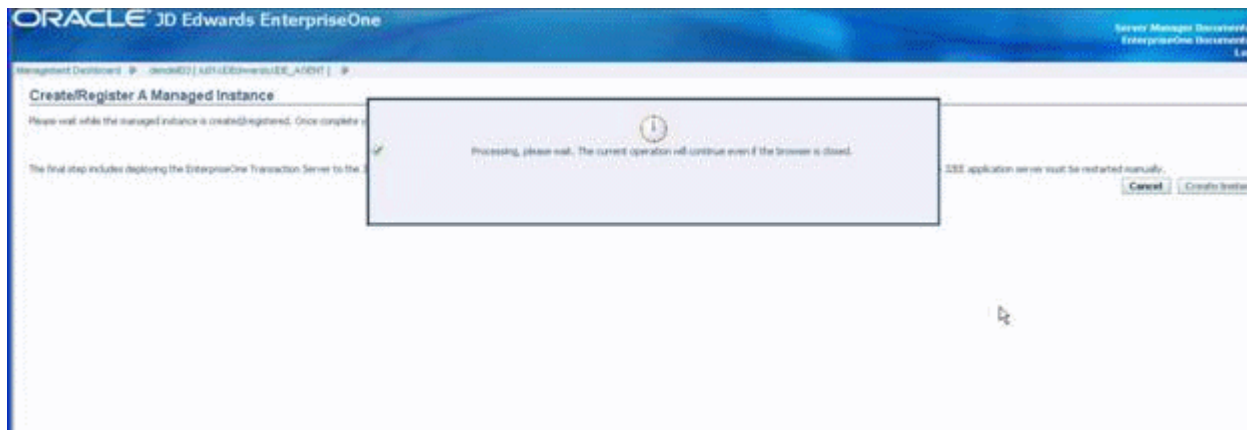
10. Click the Continue button.

11. On Create/Register A Managed Instance, Confirmation, complete these fields:

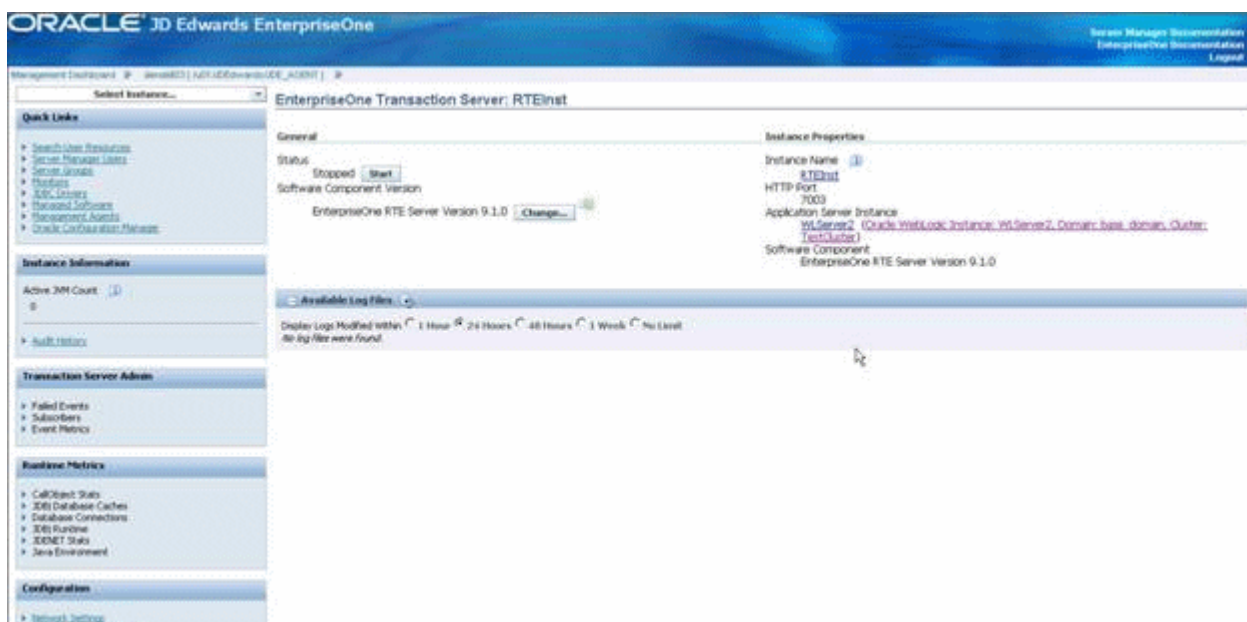
- *Bootstrap User*
Enter any value that you will use for this user.
- *Bootstrap User Password*
Enter a password.
- *Bootstrap Role*
Enter the JD Edwards EnterpriseOne role for the bootstrap user. For example, *ALL is a valid role.
- *Sequencing Behavior*
Use the pulldown to select the desired behavior.
- *Processing Concurrency*
Use the pulldown to select the desired processing concurrency value. This defines the number of concurrent event processing threads to run.
- *WebSphereMQ Initial Root*
Leave this value blank; it is not applicable to WebLogic installations.

12. Click the Continue button.

13. On Create/Register A Managed Instance, Finish, click the Create Instance button.

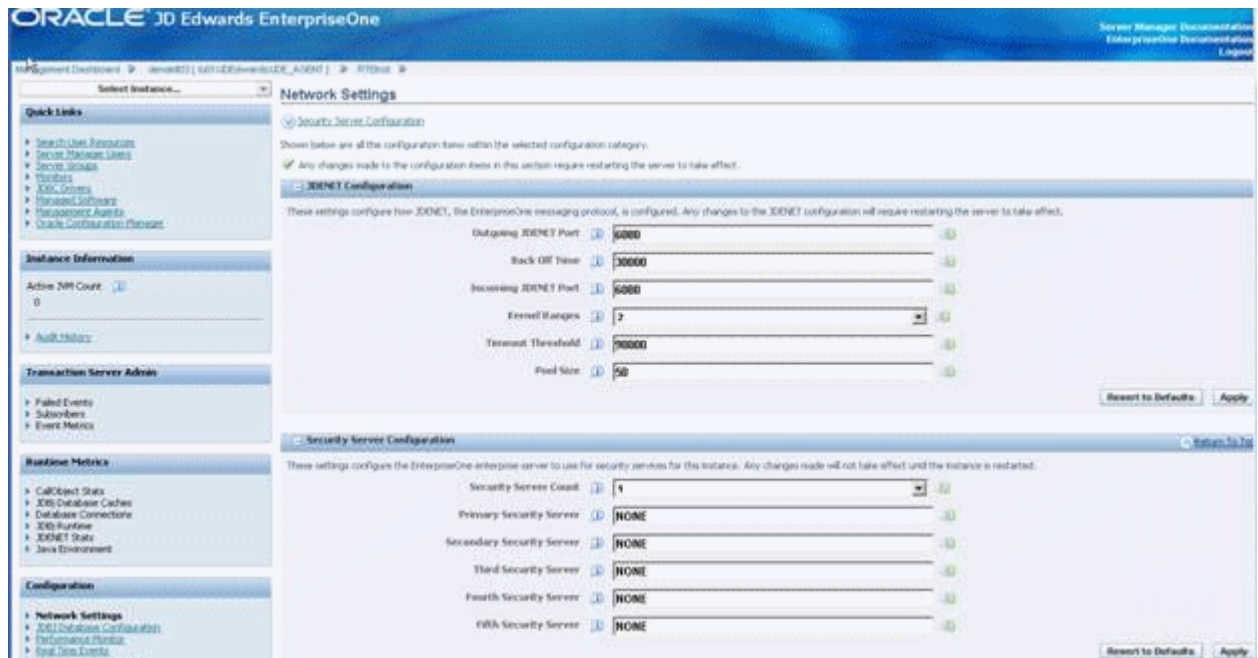


Server Manager displays a processing message.

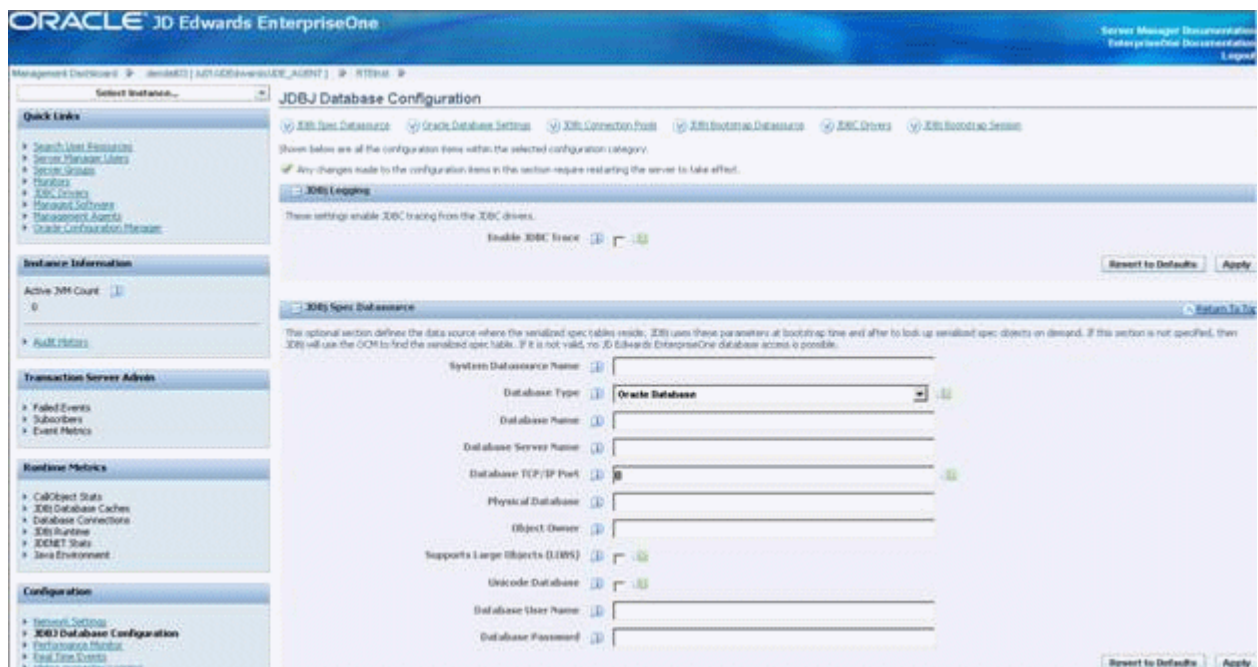


The instance screen for the RTE application displays.

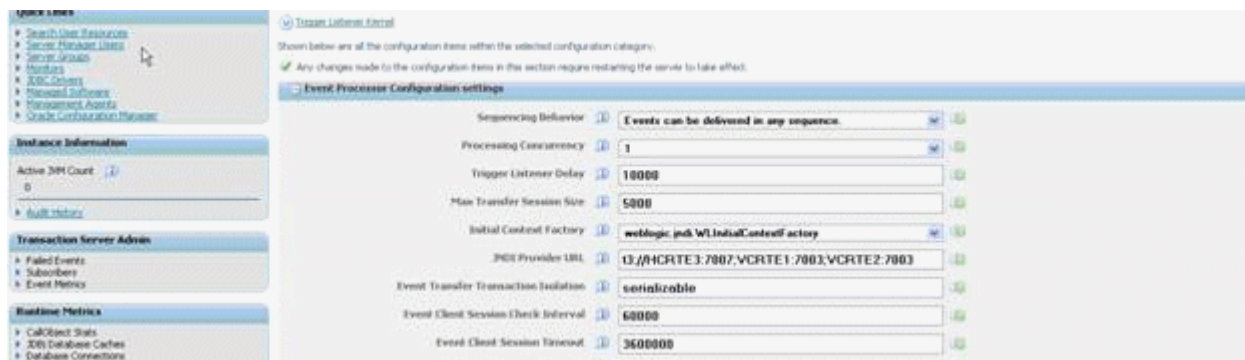
Caution: Do not attempt to start the instance at this time. You must continue with the setup instructions in this guide until you are explicitly instructed to start the instance.



14. In the Configuration section, click Network Settings and complete the fields as applicable for your environment.



15. In the Configuration section, click JDBJ Database Configuration and complete the fields as applicable for your environment.



16. In the Configuration section, click Event Processor Configuration settings and confirm the fields are valid for your installation.

Caution: Do not attempt to start the instance at this time. You must continue with the setup instructions in this guide until you are explicitly instructed to start the instance.

H.10 Using JDBC Persistent Stores

Follow the steps below to modify the installed configuration for using JDBC Persistent Stores instead of FileStore. In order to use horizontal clusters, it is mandatory to use JDBC Persistent Stores for JMS Servers.

Caution: Do not attempt to start the Transaction Server RTE instance or any other managed servers at this time. You must continue with the setup instructions in this guide until you are explicitly instructed to start an instance or managed server.

These instructions assume that you've already created a database to serve as a persistent store.

This chapter discusses these topics:

- [Section H.11, "Create JDBC Datasources"](#)
- [Section H.12, "Create JDBC Persistent Stores"](#)
- [Section H.13, "Associate a JMS Server with a Persistent Store"](#)

H.11 Create JDBC Datasources

Use this procedure to create JDBC Datasources.

1. Logon to the WebLogic Administration Console.
2. Go to Services > Data Sources.

Summary of JDBC Data Sources - JDE01 - WLS Co...

ORACLE WebLogic Server® Administration Console

Home Log Out Preferences Record Help

Home > Summary of Services > Summary of JDBC Data Sources

Summary of JDBC Data Sources

Configuration Monitoring

A JDBC data source is an object bound to the JNDI tree that provides database connectivity through a pool of X then borrow a database connection from a data source.

This page summarizes the JDBC data source objects that have been created in this domain.

Customize this table

Data Sources (Filtered - More Columns Exist)

Click the **Lock & Edit** button in the Change Center to activate all the buttons on this page.

New Delete

	Name	Type	JNDI Name
There are no items to display			

New Delete

- On Summary of JDBC Data Sources, click the **Lock & Edit** button.

Customize this table

Data Sources (Filtered - More Columns Exist)

New Delete

- Generic Data Source
- GridLink Data Source
- Multi Data Source

New Delete

- On Summary of JDBC Data Sources, in the **Customize this table** section, use the **New** pulldown to select **Generic Data Source**.

The screenshot shows a Java Swing window titled "Create a New JDBC Data Source". At the top, there are four buttons: "Back", "Next", "Finish", and "Cancel". Below the buttons is a section titled "JDBC Data Source Properties". A text label states: "The following properties will be used to identify your new JDBC data source." followed by a note: "* Indicates required fields". The first question is "What would you like to name your new JDBC data source?". Below it is a label "* Name:" with a required field icon (a yellow circle with an exclamation mark). To the right is a text input field containing "JDBC Data Source-TEST". The second question is "What JNDI name would you like to assign to your new JDBC Data Source?". Below it is a label "JNDI Name:" with a required field icon. To the right is a text area containing "TEST". The third question is "What database type would you like to select?". Below it is a label "Database Type:" and a dropdown menu currently showing "MS SQL Server". At the bottom of the window are four buttons: "Back", "Next", "Finish", and "Cancel".

5. On Create a New JDBC Data Source, JDBC Data Source Properties, complete these fields:
 - *Name*
Enter a name for the JDBC data source.
 - *JNDI Name*
Enter a name for the JNDI.
 - *Database Type*
Use the pulldown to select your database type.
6. Click the **Next** button.

Create a New JDBC Data Source

Back Next Finish Cancel

JDBC Data Source Properties

The following properties will be used to identify your new JDBC data source.

Database Type: MS SQL Server

What database driver would you like to use to create database connections? Note: * indicates that the driver is explicitly supported by Oracle WebLogic Server.

Database Driver: *Oracle's MS SQL Server Driver (Type 4) Versions: 7.0 and later

Back Next Finish Cancel

7. On Create a New JDBC Data Source, use the pulldown to select the driver appropriate for your database.

Note: For Oracle databases, you must select the thin driver. You cannot use an XA driver.

8. Click the **Next** button.

Create a New JDBC Data Source

Back Next Finish Cancel

Transaction Options

You have selected non-XA JDBC driver to create database connection in your new data source.

Does this data source support global transactions? If yes, please choose the transaction protocol for this data source.

☐ Supports Global Transactions

Select this option if you want to enable non-XA JDBC connections from the data source to participate in global transactions using the *Logging Last Resource (LLR)* transaction optimization. Recommended in place of Emulate Two-Phase Commit.

☒ Logging Last Resource

Select this option if you want to enable non-XA JDBC connections from the data source to emulate participation in global transactions using JTA. Select this option only if your application can tolerate heuristic conditions.

☐ Emulate Two-Phase Commit

Select this option if you want to enable non-XA JDBC connections from the data source to participate in global transactions using the one-phase commit transaction processing. With this option, no other resources can participate in the global transaction.

☒ One-Phase Commit

Back Next Finish Cancel

9. On Create a New JDBC Data Source, Transaction Options, *deselect* this checkbox:
Supports Global Transactions
10. Click the **Next** button.

The screenshot shows a Java Swing dialog box titled "Create a New JDBC Data Source". At the top, there are four buttons: "Back", "Next", "Finish", and "Cancel". Below the title bar, the section "Connection Properties" is active, with the instruction "Define Connection Properties." Below this, there are five questions, each followed by a text input field:

- Question: "What is the name of the database you would like to connect to?"
Field: "Database Name:" with the value "RTECLUSTER".
- Question: "What is the name or IP address of the database server?"
Field: "Host Name:" with the value "TOOLS910E".
- Question: "What is the port on the database server used to connect to the database?"
Field: "Port:" with the value "1433".
- Question: "What database account user name do you want to use to create database connections?"
Field: "Database User Name:" with the value "RTECLUSTER".
- Question: "What is the database account password to use to create database connections?"
Field: "Password:" with masked characters (dots). Below it is a "Confirm Password:" field, also with masked characters.

At the bottom of the dialog, there are four buttons: "Back", "Next", "Finish", and "Cancel".

11. On Create a New JDBC Data Source, Connection Properties, complete these fields:

- *Database Name*
Enter a name for the database to which you want to connect.
- *Host Name*
Enter the name or IP address of the database server.
- *Port*
Enter the port on the database server which is used to connect.
- *Database User Name*
Enter the database account you want to use to create the database connections.
- *Password*
Enter the database account password to use to create database connections.
- *Confirm Password*
Confirm the database account password.

12. Click the **Next** button.

Messages
✓ Connection test succeeded.

Create a New JDBC Data Source

Test Configuration Back Next Finish Cancel

Test Database Connection

Test the database availability and the connection properties you provided.

What is the full package name of JDBC driver class used to create database connections in the connection pool?
(Note that this driver class must be in the classpath of any server to which it is deployed.)

Driver Class Name:

What is the URL of the database to connect to? The format of the URL varies by JDBC driver.

URL:

What database account user name do you want to use to create database connections?

Database User Name:

What is the database account password to use to create database connections?
(Note: for secure password management, enter the password in the Password field instead of the Properties field below)

Password:

Confirm Password:

13. On Create a New JDBC Data Source, Test Database Connection, click the **Test Configuration** button.
14. Ensure you receive this message: **connection test succeeded**
15. Click the **Next** button.

Create a New JDBC Data Source

Back Next Finish Cancel

Select Targets

You can select one or more targets to deploy your new JDBC data source. If you don't select a target, the data source will be created but not deployed. You will need to deploy the data source at a later time.

Servers

- ☐ AdminServer
- ☐ BSSVT1
- ☐ test2

Clusters

- ☒ RCV1
 - ☒ All servers in the cluster
 - ☐ Part of the cluster
 - ☐ RTEC1
 - ☐ RTEC2

Back Next Finish Cancel

16. On Summary of JDBC Data Sources, in the list of Data Sources (Filtered - More Columns Exist), select the checkboxes for all servers in the cluster.

Summary of JDBC Data Sources

Configuration Monitoring

A JDBC data source is an object bound to the JNDI tree that provides database connectivity through a pool of JDBC connections. Applications can look up a data source on the JNDI tree and then borrow a database connection from a data source.

This page summarizes the JDBC data source objects that have been created in this domain.

[Customize this table](#)

Data Sources (Filtered - More Columns Exist)

Click the **Lock & Edit** button in the Change Center to activate all the buttons on this page.

Name	Type	JNDI Name	Targets
<input type="checkbox"/> JDBC Data Source-RTEC	Generic	RTEC	RCV1

Showing 1 to 1 of 1 Previous Next

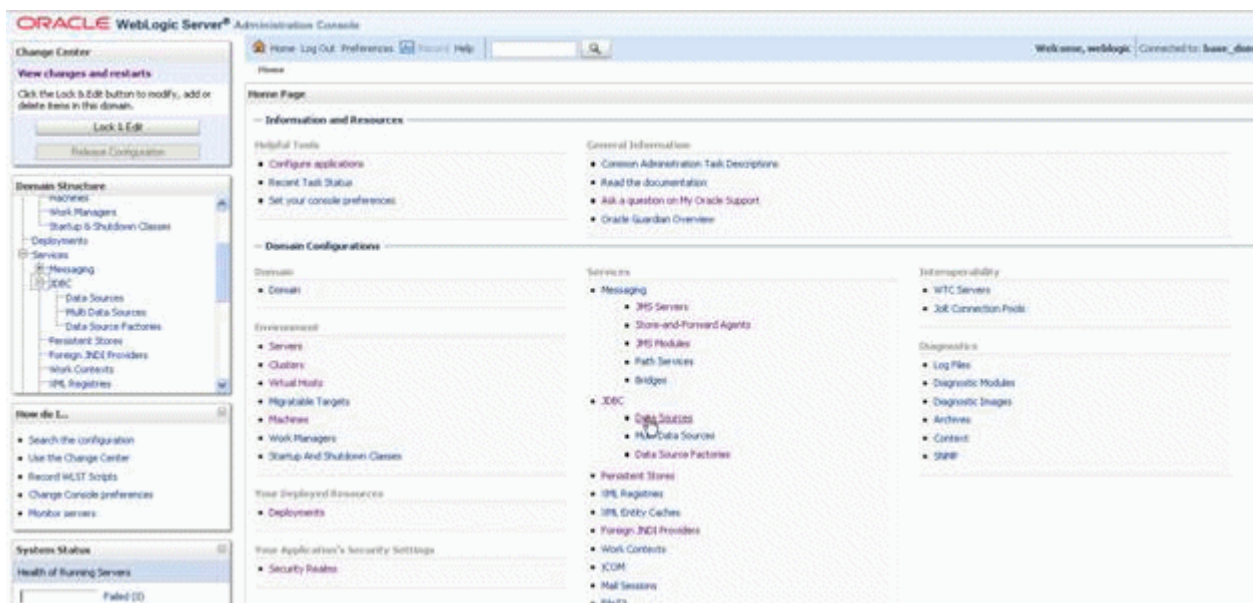
17. Click the **Finish** button at the bottom of the page.
18. Repeat the applicable steps in this procedure to add additional JDBC data sources.
19. Click **Activate changes**.

H.12 Create JDBC Persistent Stores

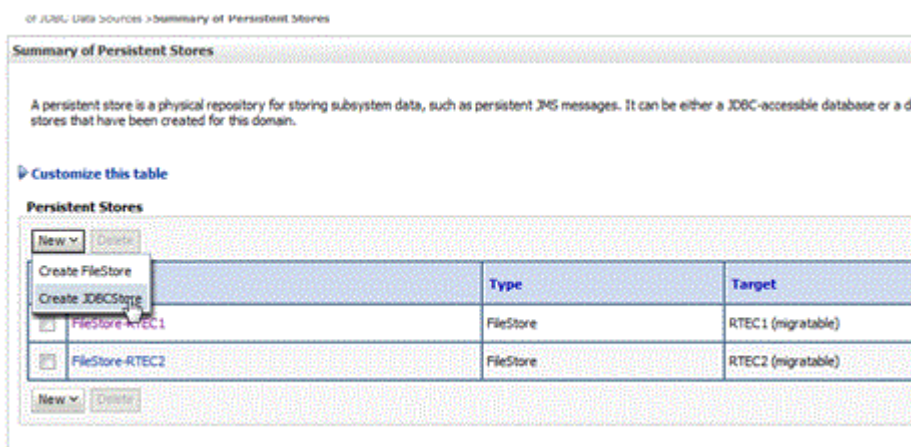
Server Manager deploys the RTE solution with file persistent stores already created. This works for vertical clusters, but should be changed to JDBC type persistent stores for horizontal clusters.

Use this procedure to create the necessary JDBC persistent stores.

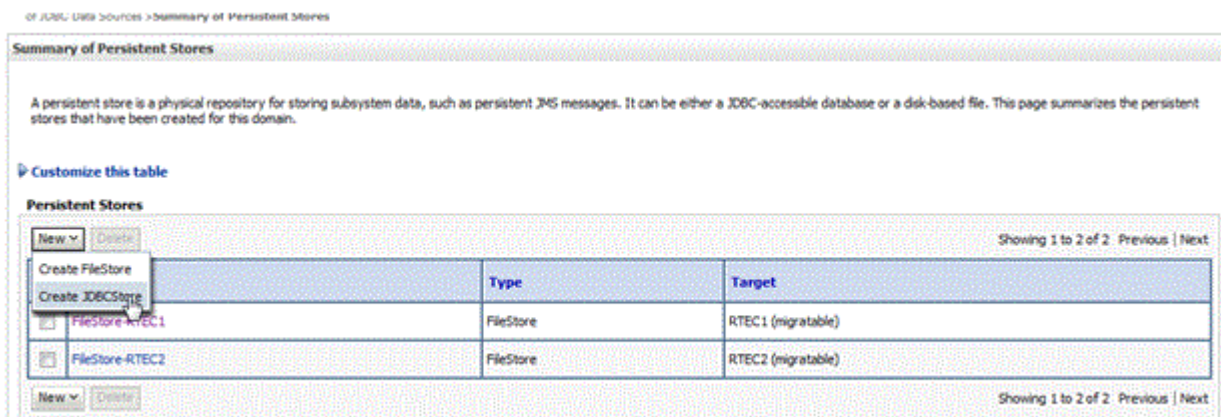
1. Go to **Services > Persistent Stores**.



2. On Summary of Persistent Stores, click the **Lock & Edit** Button.



3. On Summary of Persistent Stores, in the **Customize this table** section, click the **New** pulldown and on the flyout choose **Create JDBCStore**.



4. On Settings for JDBCStore, complete these fields:

■ *Name*

Enter a name for the JDBC store. The name must be unique within the WebLogic Server instance or its cluster.

■ *Target*

Use the pulldown to select the migratable server. The list will show of all WebLogic Server instances that have been defined in the current domain and are therefore candidates for hosting this JDBC store.

■ *Data Source*

Use the pulldown to select the JDBC data source used by this JDBC store to access its backing table.

Caution: The list will only show JDBC datasources that:

- Do not use global transactions.
 - Do not use a non XA JDBC driver.
 - Are not configured to enable Logging Last Resource
 - Are not configured to Emulate Two-Phase Commit
-
-

■ *Prefix Name*

Leave this field blank.

5. Click the Save button.

6. Click the OK button.

7. Repeat the applicable steps to add additional persistent stores for additional JMS servers.

8. Click the Activate Changes button.

When you complete this procedure, your list of server will be similar to this:

JMS Servers (Filtered - More Columns Exist)

Click the *Lock & Edit* button in the Change Center to activate all the buttons on this page.

<input type="checkbox"/>	Name	Persistent Store	Target	Current Server	Health
<input type="checkbox"/>	JMServer-RTEC1	JDBCStore-RTEC1	RTEC1 (migratable)	RTEC1	
<input type="checkbox"/>	JMServer-RTEC2	JDBCStore-RTEC2	RTEC2 (migratable)	RTEC2	

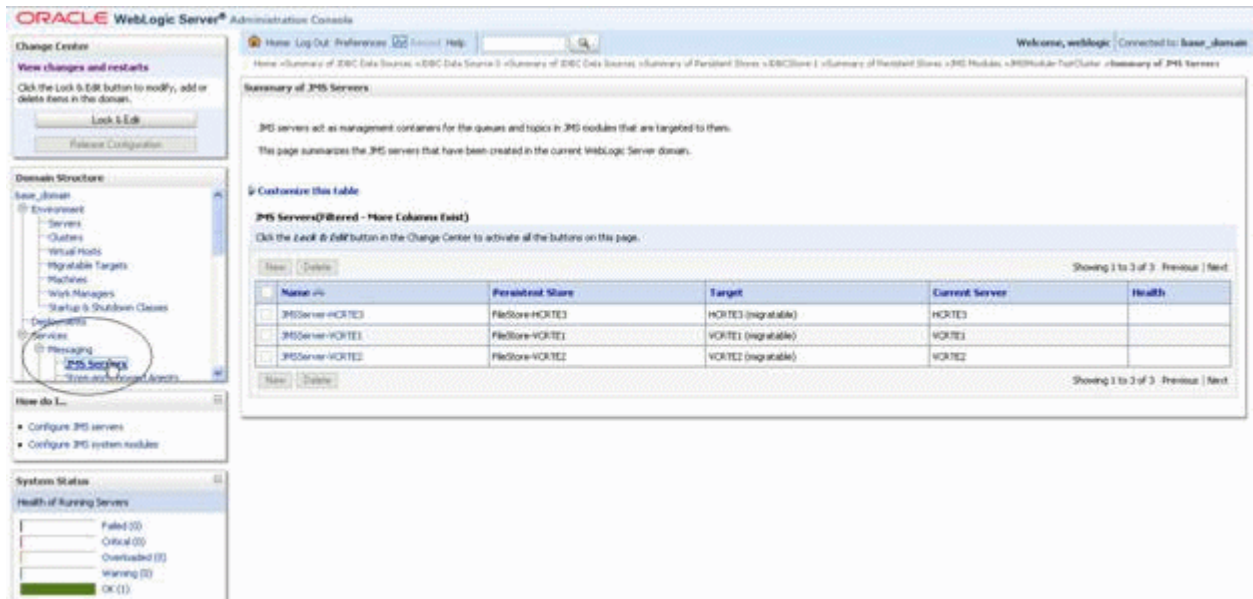
Showing 1 to 2 of 2 Previous | Next

H.13 Associate a JMS Server with a Persistent Store

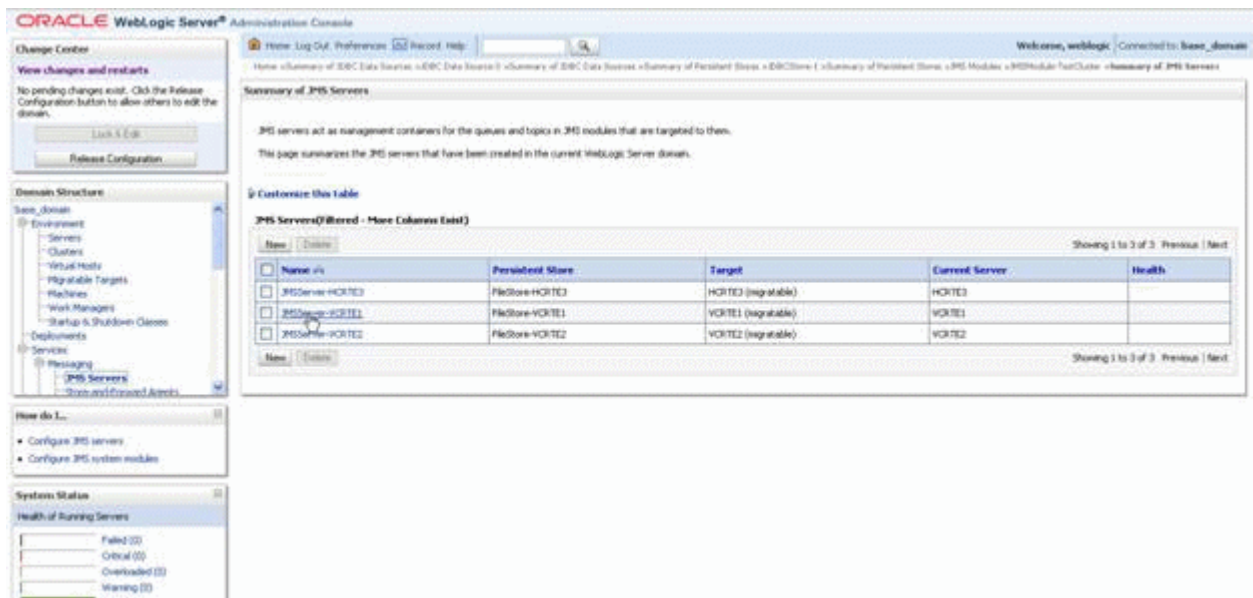
Server Manager by default associates the JMS Servers with file persistent stores. For using JDBC store or using horizontal cluster, you must associate the JMS Servers with JDBC persistent stores.

Use this procedure to associate a JMS Server with a persistent store:

1. Go to Services > Messaging > JMS Servers.



2. On Summary of JMS Server, click the Lock & Edit button at the bottom of the page.



3. On Summary of JMS Server, in the JMS Servers (Filtered - More Columns Exist), click on the JMS Server that you want to associate with a persistent store.

The screenshot shows the 'Configuration' window for a JMS server. The 'General' tab is selected. The 'Name' field is 'JMServer-VOTEL'. The 'Persistent Store' dropdown is set to 'JDBCStore-1'. The 'Paging Directory' is empty. The 'Message Buffer Size' is '-1'. The 'Hosting Temporary Destinations' checkbox is checked. The 'Module Containing Temporary Templates' is 'none'. The 'Temporary Template Name' is empty. The 'Expiration Scan Interval' is '30'. The 'Save' button is at the bottom right.

4. Click on the General Tab and complete these fields:
 - *Persistent Store*
Use the pulldown to select the JDBC Store to which you want to associate.
 - *Paging Directory*
Accept the default value.
 - *Message Buffer Size*
Accept the default value.
 - *Hosting Temporary Destinations*
Ensure the checkbox for this option is selected.
 - *Module Containing Temporary Templates*
Accept the default value.
 - *Temporary Template Name*
Accept the default value.
 - *Expiration Scan Interval*
Accept the default value.
5. Click the Save button.
6. Click the Activate Changes button.
7. Repeat the applicable steps to add JMS Servers and persistent stores.

When you complete this procedure, your list of server will be similar to this:

Server Manager creates the necessary file structure on the node that the WebLogic Administration Console resides on. However, for any horizontal cluster nodes, these files should be transferred manually. On Windows Machines, this should be done through Windows Explorer. For Unix or Linux machines, this should be done through FTP. The steps below use Linux/FTP as an example.

- Services

Messaging

IMB Servers

Outbound Forward Agents

How do I...

 - Configure startup arguments for Managed Servers
 - Start Managed Servers from the Administration Console
 - Shut down a server instance

System Status

Health of Running Servers

<div></div>	Failed (0)
<div></div>	Critical (0)
<div></div>	Overloaded (0)
<div></div>	Warning (0)
<div></div>	OK (1)

Root Directory:

Class Path:

Arguments:

-DDefaultHost=ipd11j000swsdc300

ADMT/Targets/DTGInstm

/conf3g

-XX:MaxPermSize=100m

-Xms512m

-Xmx1024m

Security Policy File:

User Name:

Password:

Confirm Password:

Save

The directory that this server uses as its root directory. The directory must be on the computer that hosts the Node Manager. If you do not specify a Root Directory value, the default directory is used by default.

More Info...

The classpath (path on the machine running Node Manager) to use when starting this server.

More Info...

The arguments to use when starting this server.

More Info...

The security policy file (directory and filename on the machine running Node Manager) to use when starting this server.

More Info...

The user name to use when logging this server.

More Info...

The password of the username used to host the server and perform server health monitoring.

More Info...

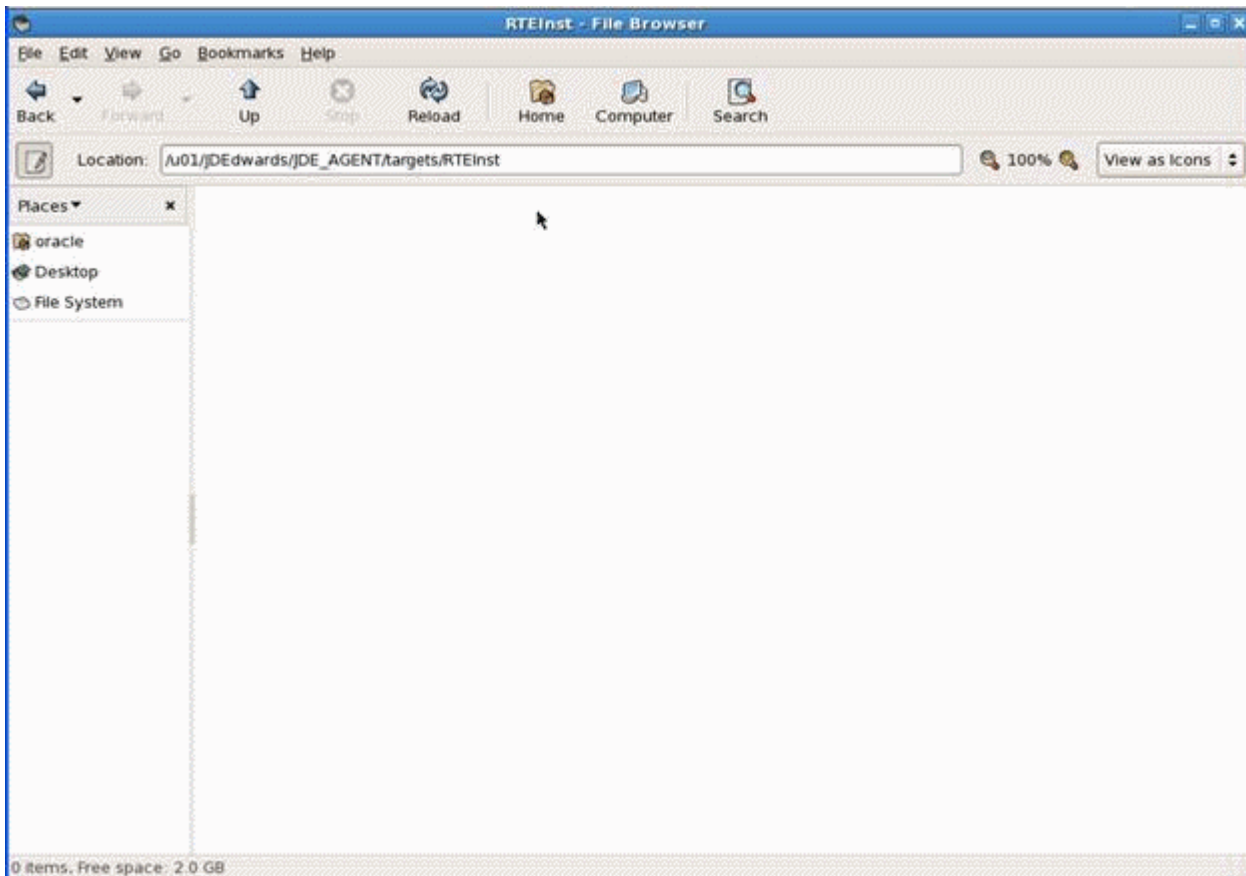
- Clustering for the JD Edwards Transaction Server on WebLogic Server H-31

/u01/JDEdwards/JDE_AGENT/targets/RTEInst/config

This is the directory that is created on the horizontal node and where the necessary files must be copied.

- d. On the target machine, you must pre-create the necessary directory structure.

For example:



- e. Copy the config files from the vertical node machine to the horizontal node machine.

In the below example, the vertical node machine is dendell03 and the horizontal node machine is dnhpb112.

Note: If you are using UNIX or Linux machines, ensure you have the correct permissions for the new directory. For example:

```
chmod 777 -R JDEdwards
```

