Oracle Financial Services Inline Processing Engine

Configuration Guide

Release 8.0.7.0.0

November 2018

ORACLE
FINANCIAL SERVICES

ORACLE

OFS Inline Processing Engine Configuration Guide

Document Control

Version Number	Revision Date	Change Log
1.0	Created: November 2018	Added 8.0.7.0.0 Minor Release installation procedure and configuration details.
2.0	Updated: November 2018	Updated for Doc 28952819.
3.0	Updated: January 2019	Updated for Doc 29243430.
4.0	Updated: April 2019	Updated for Doc 29531527.

Table of Contents

1 Preface 7

1.1 Summary 7
1.2 Audience 7
1.2.1 Prerequisites for the Audience 7
1.3 Related Documents 7
1.4 Conventions and Acronyms 8
1.4.1 Acronyms 8

2 About OFSAAI IPE 9

2.1 About OFSAA 9
2.2 About IPE 9

3 Configuring IPE 10

3.1 Prerequisites 10

4 Configuring IPE in Web Application Servers for Real Time Mode 11

4.1 Configuring IPE in WebLogic 11
4.1.1 Login to WebLogic Administrative Console 11
4.1.2 Configuring JMS Servers 11
4.1.3 Configuring JMS Modules 13
4.1.4 Creating Subdeployments 15
4.1.5 Creating JMS Connection Factory 17
4.1.6 Creating JMS Topic 21
4.1.7 Creating JMS Queues 25
4.1.8 Restart Weblogic Domain 28
4.2 Configuring IPE in WebSphere 29
4.2.1 Login to WebSphere 29
4.2.2 Bus Creation 29
4.2.3 Bus Member Creation 30
4.2.4 Configuring JMS Providers 33
4.2.5 Configuring JMS Connection Factory 34

4.2.6 Configuring JMS Queues 39
4.2.7 Configuring JMS Topics 45
4.2.8 RMI/IIOP Authentication Settings 48
4.2.9 Restart WebSphere Profile 50
4.3 Configuring IPE in Tomcat 50
4.3.1 Prerequisites 50
4.3.2 Configuring Tomcat 50
4.4 Configuring IPE in Kafka 52
4.4.1 Prerequisites 53
4.4.2 Configuring Kafka 53
4.4.3 Testing Kafka Configuration 55

5 Post data in JSON format to IPE 57

5.1 Prerequisites 57
5.2 Posting JSON request for REST services 57

6 Additional Configuration 59

6.1 Updating Install Properties 59
6.1.1 Updating install.properties to enable Highlights in Real-Time 59
6.1.2 IPE Caching 60
6.2 Creating ILP.ear/ ILP.war 60
6.3 Deploying ILP.ear in Weblogic 61
6.3.1 Installing ILP.ear in WebLogic using WebLogic Administrator Console. 62
6.3.2 Deploying ILP.ear in WebSphere 64
6.4 Logging 70

7 HIVE Configurations 71

7.1 Loading UDF file in HIVE 71
7.2 Creating Result Tables 71

8 Appendix A 72

8.1 Inline Processing URL of JMS 72

9 Appendix B 73

9.1 Check ports in WebSphere 73

10 Appendix C 75

10.1 Checking Target Inbound transport chain and Provider endpoints values 75

1 Preface

This Preface provides supporting information for the Oracle Financial Services Analytical Applications Infrastructure Inline Processing Configuration (OFSAAI IPE) Guide and includes the following topics:

- Summary
- Audience
- Documentation Accessibility
- Related Documents
- Conventions

1.1 Summary

You can find the latest copy of this document in the OHC library which includes all the recent additions/revisions (if any) done till date.

Before you begin the installation, ensure that you have an access to the Oracle Support Portal with the required login credentials to quickly notify us of any issues at any stage. You can obtain the login credentials by contacting Oracle Support.

1.2 Audience

Oracle Financial Services Analytical Applications Infrastructure Inline Processing Application Configuration Guide is intended for administrators and implementation consultants who are responsible for installing and maintaining the application pack components.

1.2.1 Prerequisites for the Audience

Following are the expected preparations before starting the actual installation:

The document assumes you have experience in installing Enterprise components. Basic knowledge about the Oracle Financial Services Analytical Applications Infrastructure Applications Pack components, Inline Processing Sample Application, OFSAA Architecture, UNIX commands, Database concepts and Web Server/ Web Application Server is recommended.

1.3 Related Documents

This section identifies additional documents related to OFSAA Infrastructure. You can access Oracle documentation online from the Documentation Library for OFSAAI 8.0.7.0.0 (OHC).

- *Oracle Financial Services Analytical Applications Infrastructure Inline Processing User Guide*
- *Oracle Financial Services Analytical Applications Infrastructure Inline Processing Sample Application Installation Guide*
- *Release Notes*

1.4 Conventions and Acronyms

The following conventions are used in this guide.

Conventions	Description
Bold.	Indicates Actions
`Courier New font`	Indicates Command or Query

1.4.1 Acronyms

This guide contains the following acronyms.

Acronyms	Description
IPE	Inline Processing Engine
Infodom	Information Domain
OFS AAI	Oracle Financial Services Analytical Application Infrastructure.
OFS AAAI	Oracle Financial Services Advanced Analytical Application Infrastructure.
OHC	Oracle Help Center
JSON	JavaScript Object Notation
REST	Representational State Transfer
JMS	Java Message Service

2 About OFSAAI IPE

This chapter includes the following topics:

- About OFSAA
- About IPE

2.1 About OFSAA

In today's turbulent markets, financial institutions require a better understanding of their risk-return, while strengthening competitive advantage and enhancing long-term customer value. Oracle Financial Services Analytical Applications (OFSAA) enable financial institutions to measure and meet risk adjusted performance objectives, cultivate a risk management culture through transparency, lower the costs of compliance and regulation, and improve insight into customer behavior.

OFSAA uses industry-leading analytical methods, shared data model and applications architecture to enable integrated risk management, performance management, customer insight, and compliance management. OFSAA actively incorporates risk into decision making, enables to achieve a consistent view of performance, promote a transparent risk management culture, and provide pervasive intelligence.

Oracle Financial Services Analytical Applications delivers a comprehensive, integrated suite of financial services analytical applications for both banking and insurance domain.

2.2 About IPE

This guide provides step by step instructions for performing Inline Processing Configuration process actions. Inline Processing builds a scoring mechanism for activity data like transactions in real time or batch mode. For example, this capability enables you to identify fraud events earlier, avert more losses, and minimize customer service and retention issues.

3 Configuring IPE

This chapter discusses the prerequisite instructions required to configure IPE.

3.1 Prerequisites

The following prerequisite configurations must be verified before installation:

- A user must be created and mapped with the IPEADMN (Inline Processing Admin Group) user group. To create a user, refer *Oracle Financial Services Advanced Analytics Applications Infrastructure User Guide* on OHC.
- The IPEADMIN user group must be mapped with Infodom.
- Connection Pooling and Data Source must be created for the following schemas:
  - Config Schema with Data Source name as jdbc/FICMASTER
  - IPE Atomic Schema with Data Source name as jdbc/<INFODOM NAME>

> **NOTE** jdbc/<INFODOM NAME> should be pointed to metadom. For more information, refer the section on **Information Domain** in *Oracle Financial Services Advanced Analytics Applications Infrastructure User Guide* on OHC.

  - IPE Atomic schema with Data Source name as jdbc/<INFODOM NAME>CNF

> **NOTE** jdbc/<INFODOM NAME>CNF is required only for real time processing

- For more information, refer to section *Configuring Resource Reference* of the *OFS AAAI Application Pack Installation and Configuration Guide.*
- **Oracle Database Patches**: Ensure that the patches mentioned in the section *Hardware and Software Properties* of the *OFS AAAI Application Pack Installation and Configuration Guide 8.0.2.0.0* are applied.
  - The WebSphere JDBC Providers should point to the oracle driver file path where the patch is installed.

4 Configuring IPE in Web Application Servers for Real Time Mode

This section explains the details about configuring the Web Application Servers.

This section includes the following topics:

- Configuring IPE in WebLogic
- Configuring IPE in WebSphere
- Configuring IPE in Tomcat
- Configuring IPE in Kafka

4.1 Configuring IPE in WebLogic

To configure IPE in WebLogic, follow these topics:

- Login to WebLogic Administrative console
- Configuring JMS Servers
- Configuring JMS Modules
- Creating Subdeployments
- Creating JMS Connection Factory
- Creating JMS Topic
- Creating JMS Queues

4.1.1 Login to WebLogic Administrative Console

To login to the WebLogic Administrative Console, follow these steps:

1. Open the following URL in the browser window: http://<ipaddress>:<administrative console port>/console. (https if SSL is enabled). The Welcome window is displayed.
2. Login with the Administrator Username and Password.

4.1.2 Configuring JMS Servers

To configure JMS Servers, follow these steps:

1. In the Domain Structure LHS menu, click **+** to expand **Services**.
2. Click **+** to expand **Messaging**. The *WebLogic Server Administration Console* page is displayed.
3. Select **JMS Servers**. The *Summary of JMS Servers* page is displayed.

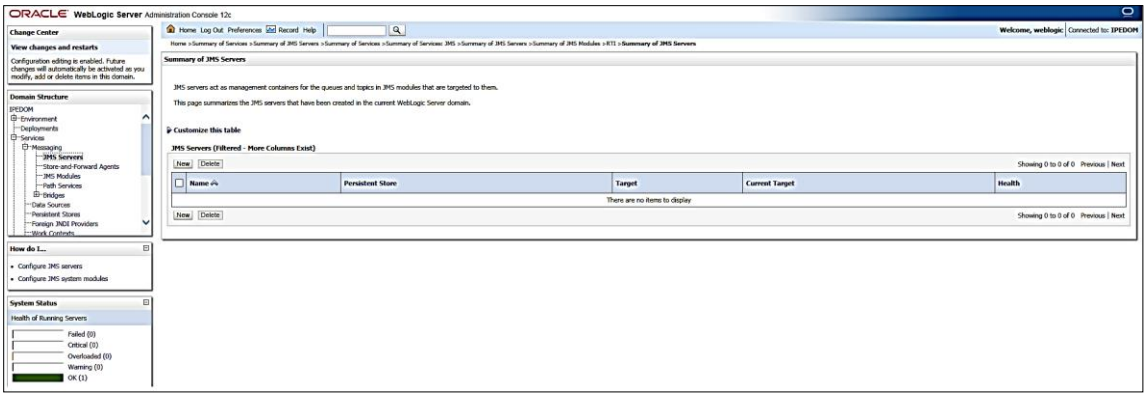


Figure 1: Summary of JMS Server

4. Click **New**. The *Create a New JMS Server* page is displayed.

Figure 2: Create a New JMS Server

5. Enter the name as **RTIServer** under JMS Server Properties.
6. Click **Next**. The *Select Targets* section is displayed.

Figure 3: Create a New JMS Server – Select Targets

7. Select the Target as **AdminServer**.
8. Click **Finish**.

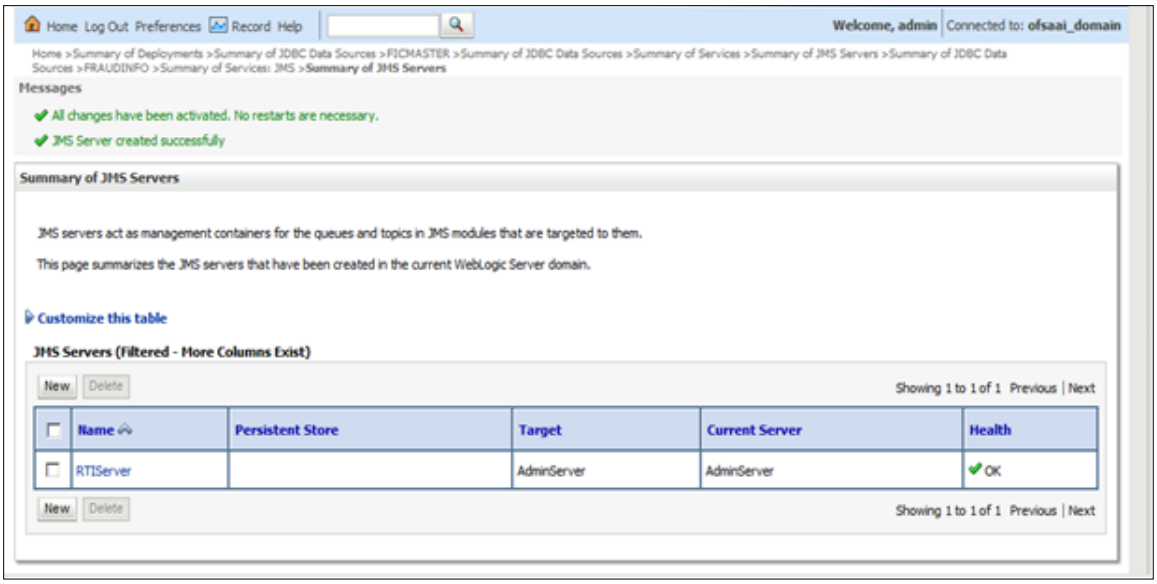


Figure 4: Summary of JMS Servers

9. The following confirmation message is displayed *JMS Server is created successfully*.

4.1.3 Configuring JMS Modules

To configure JMS Modules, follow these steps:

1. In the Domain Structure LHS menu, click **+** to expand **Services**.
2. Click **+** to expand **Messaging**.
3. Click **JMS Modules**. The *JMS Module* screen is displayed.

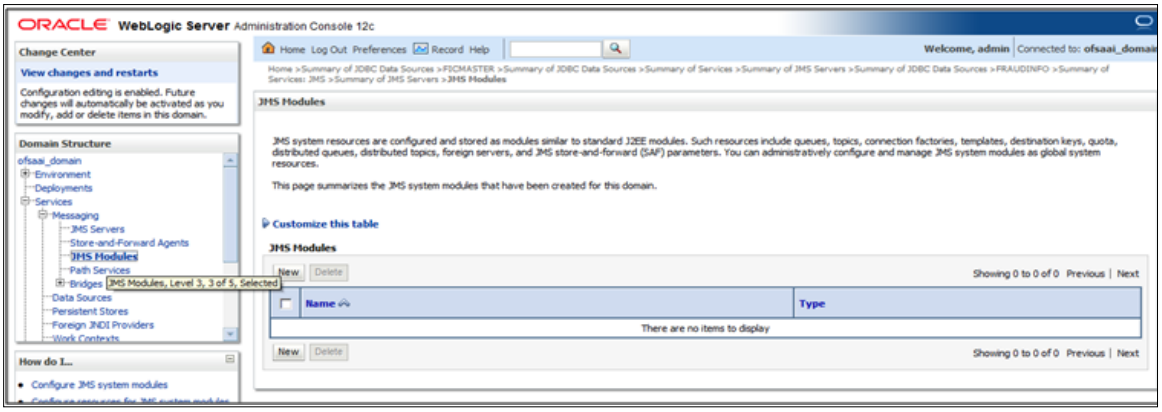
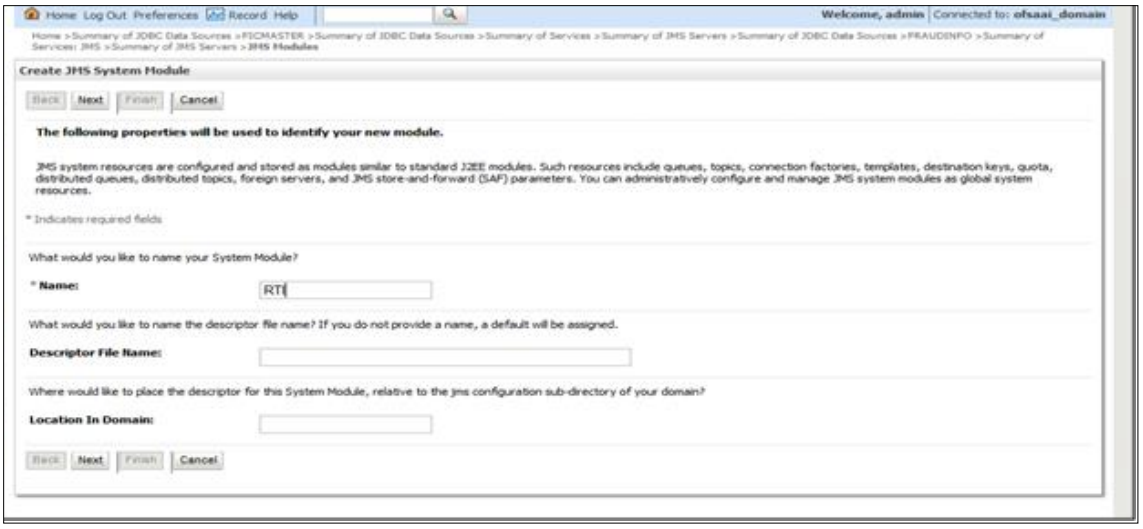


Figure 5: JMS Module

4. Click **New**. The *Create JMS System Module* page is displayed.



5. Enter the name as **RTI**.
6. Click **Next**. The *Create JMS System Module* page is displayed.

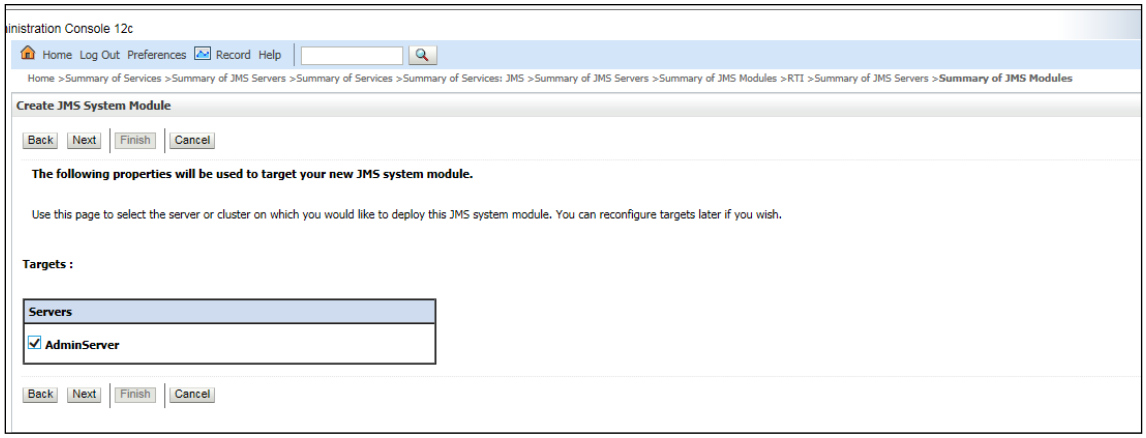


Figure 6: Create JMS System Module

7. Check the **AdminServer** in the *Servers* section.
8. Click **Next**.

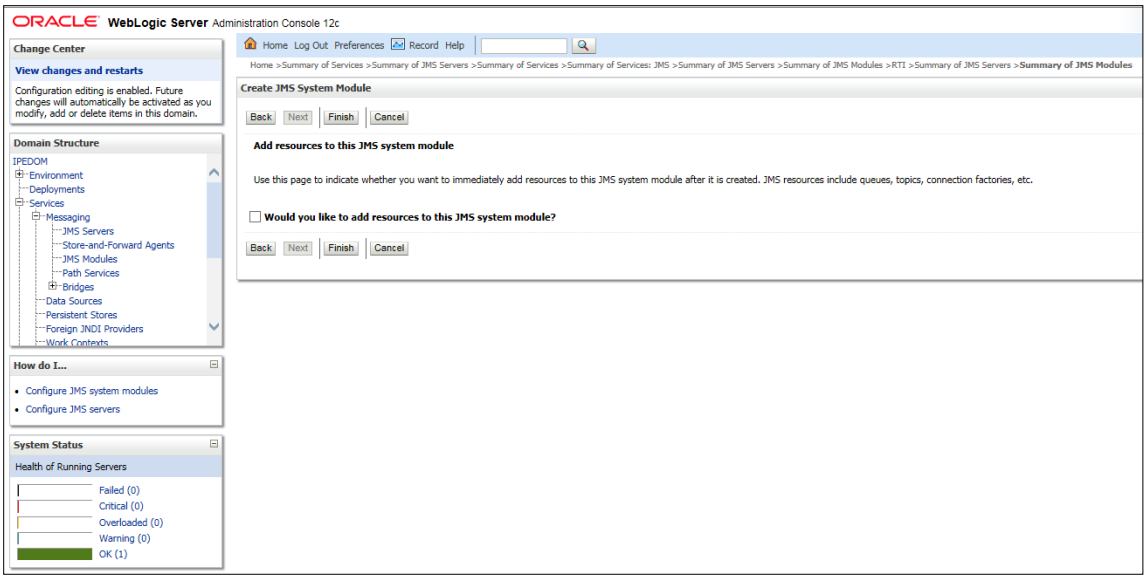


Figure 7: Create JMS System Module

9. Click **Finish**.

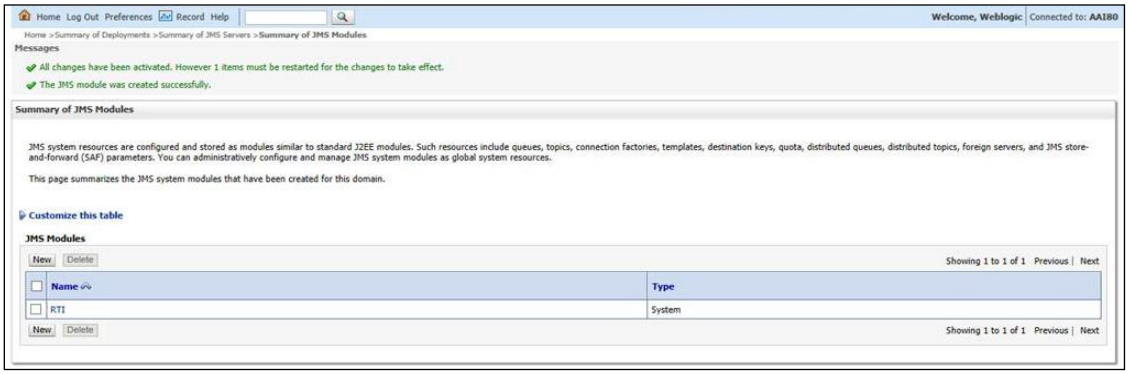


Figure 8: Summary of JMS Modules

10. The following confirmation message is displayed: *JMS Module is created successfully*.

4.1.4 Creating Subdeployments

This section discusses the following Subdeployments which are to be created

- Creating RTI Deploy
- Creating RTISubdeploy

4.1.4.1 Creating RTI Deploy

To create **RTI Deploy** subdeployment, follow these steps:

1. In the Domain Structure LHS menu, click **+** to expand **Services**.
2. Click **+** to expand **Messaging**.
3. Click **JMS Modules**. The *JMS Module* screen is displayed.

4. Click JMS Module **RTI**. The *Settings for RTI* screen is displayed.
5. Click the **Subdeployments** tab.

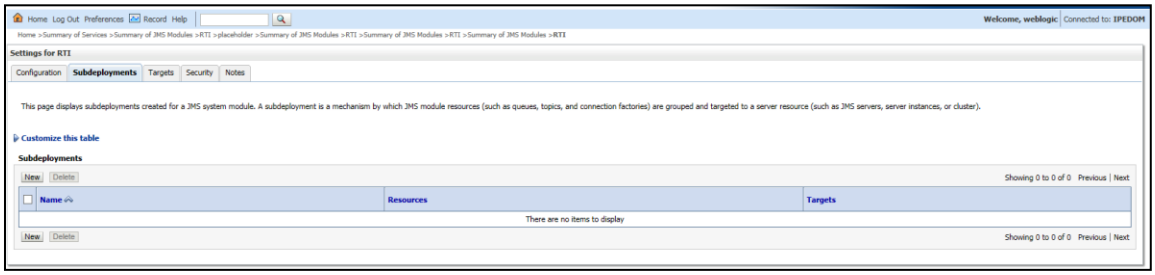


Figure 9: Setting for RTI

6. Click **New**. The *Create a New Subdeployment* screen is displayed.

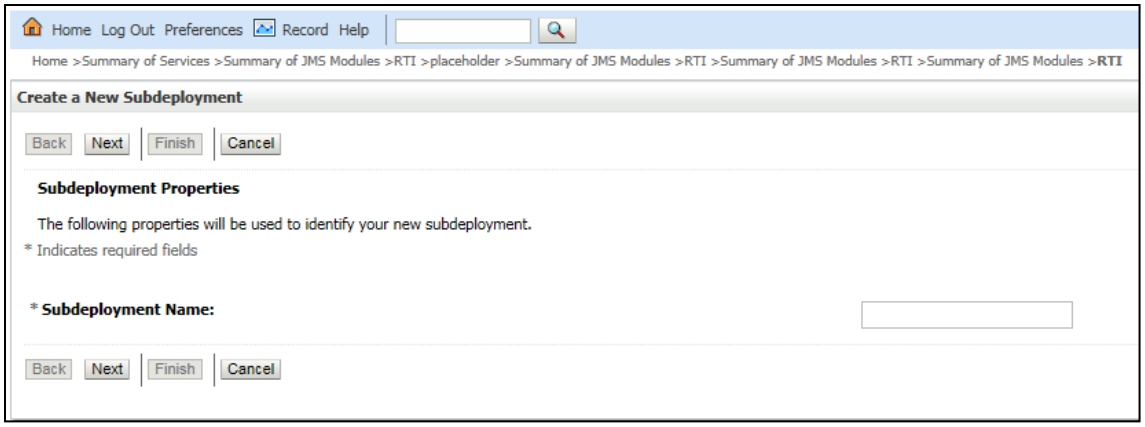


Figure 10: Create a New Subdeployment

7. Enter the Subdeployment Name as **RTI Deploy**.
8. Click **Next**.

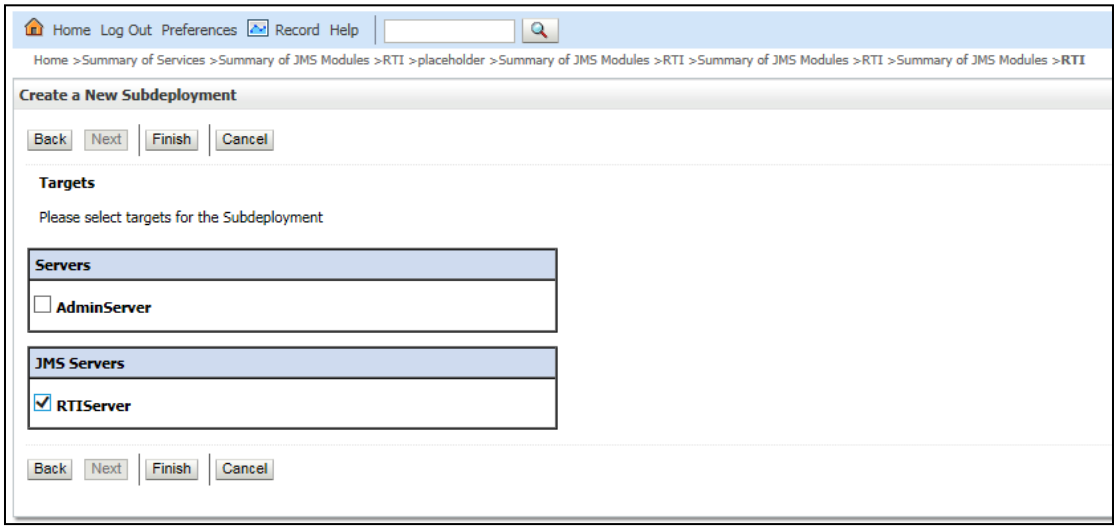


Figure 11: Create a New Subdeployment

9. Select the JMS Servers as **RTIServer**.
10. Click **Finish**. The following confirmation message is displayed: *Subdeployment is created successfully*.

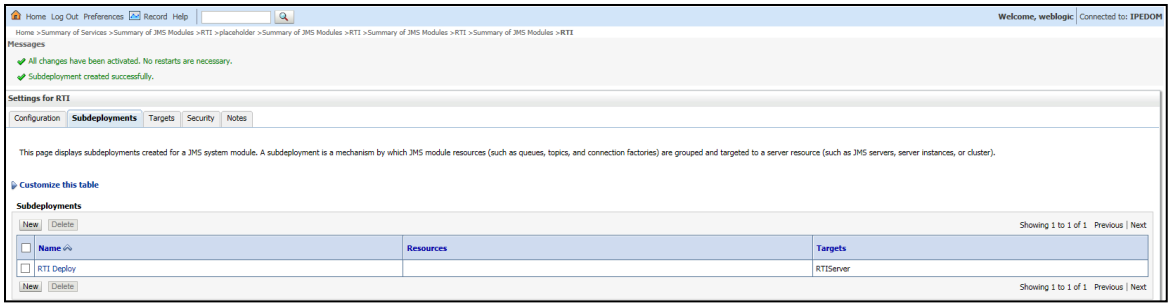


Figure 12: Settings for RTI

4.1.4.2 Creating RTISubdeploy

To create **RTISubdeploy** subdeployment, follow these steps:

1. Repeat Steps 1 - 7 from section Create RTI Deploy.
2. Enter the following details:

Table 1: Subdeployment - Field Values

Field	Value
Subdeployment Name	Enter RTISubdeploy as the name.
JMS Servers	Select RTIServer as the JMS Server.

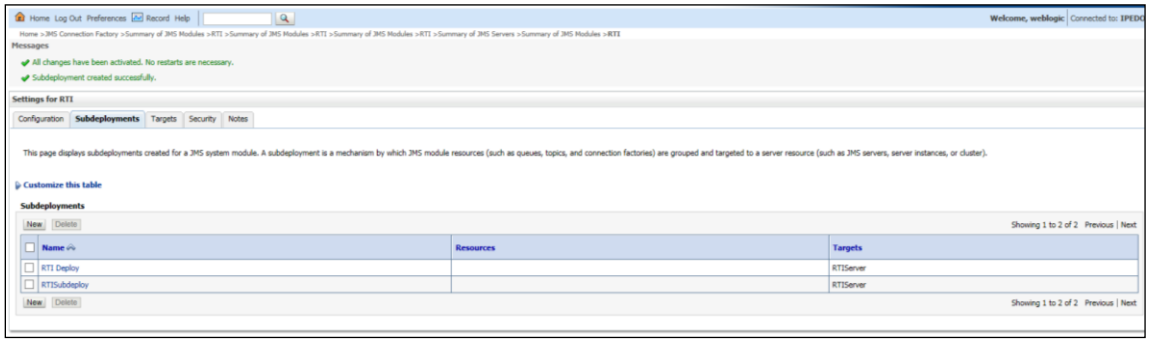


Figure 13: Subdeployments

3. The following confirmation message is displayed: *Subdeployment is created successfully*.

4.1.5 Creating JMS Connection Factory

To create JMS Connection Factories, follow these steps:

1. In the Domain Structure LHS menu, click **+** to expand **Services**.
2. Click **+** to expand **Messaging**.
3. Click **JMS Modules**. The *JMS Modules* screen is displayed.

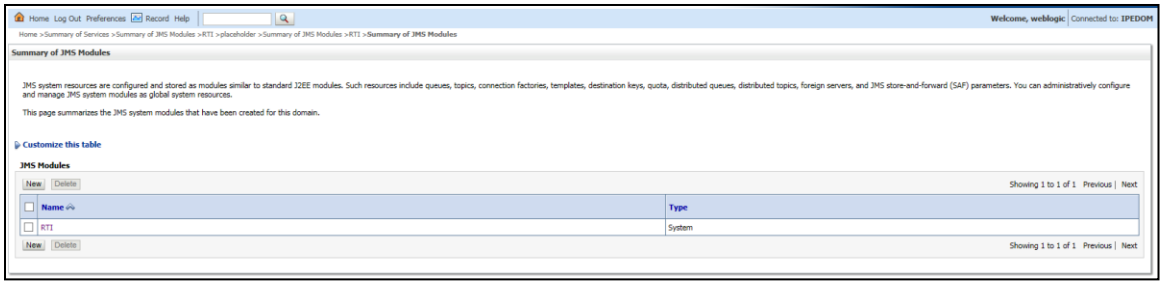


Figure 14: JMS Modules

4. Click **RTI**. The *Settings for RTI* screen is displayed.

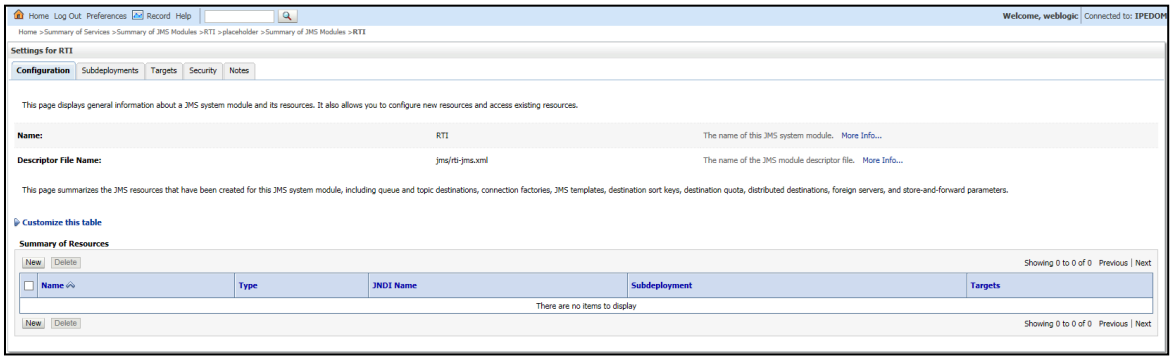


Figure 15: Settings for RTI

5. Click **New**. The *Create a New JMS System Module* screen is displayed.

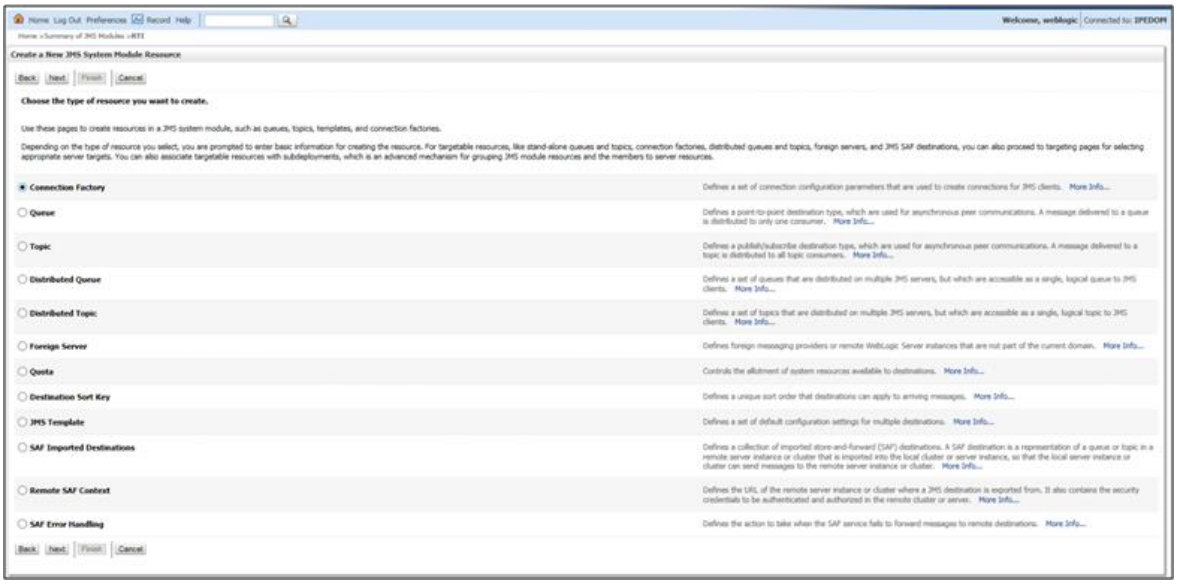


Figure 16: Create a New JMS System Module Resource

6. Select Connection Factory.
7. Click **Next**. The Create a New JMS System Module Resource screen is displayed.

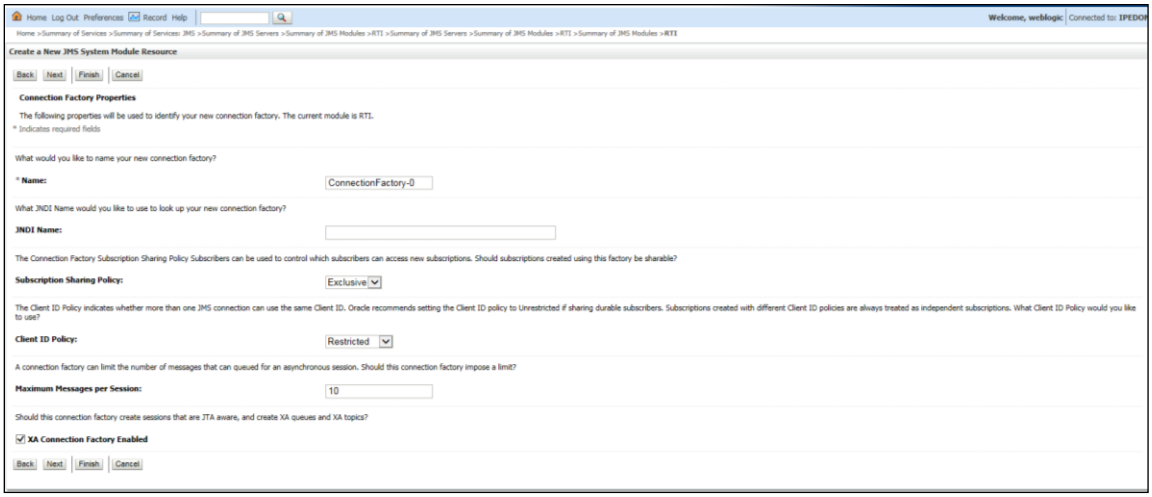


Figure 17: Create a New JMS System Module Resource

8. Enter the Name as JMS Connection Factory.
9. Click **Next**. The *Create a New JMS System Module Resource* screen with the Target section is displayed.

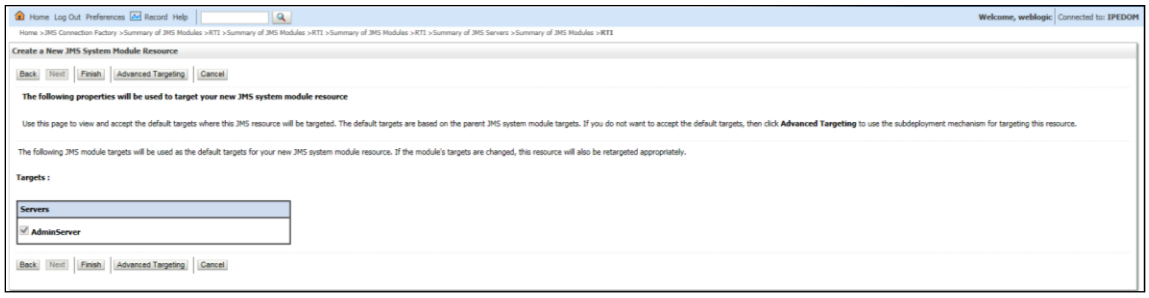


Figure 18: Create a New JMS System Module Resource - Targets

10. Select AdminServer.
11. Click **Finish**.

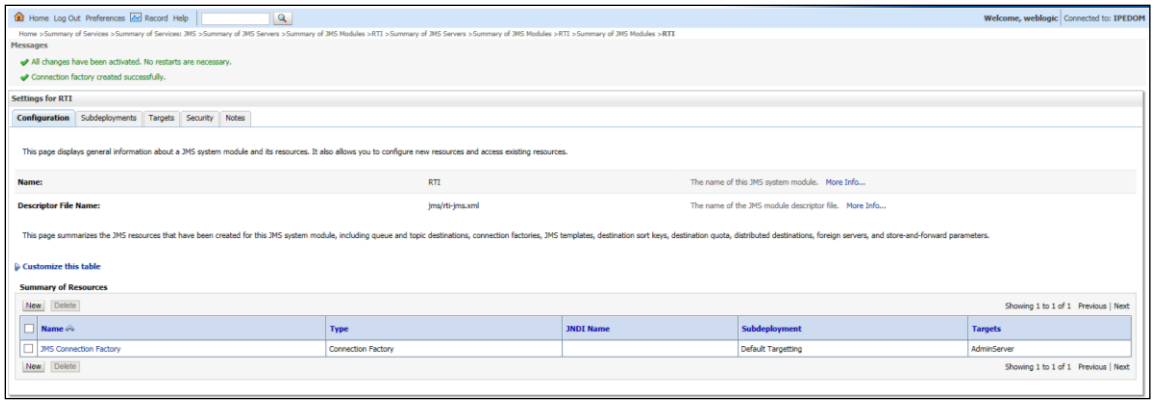


Figure 19: Settings for RTI - JMS connection Factory

12. Click **JMS Connection Factory**. The *Settings for JMS Connection Factory* screen is displayed.

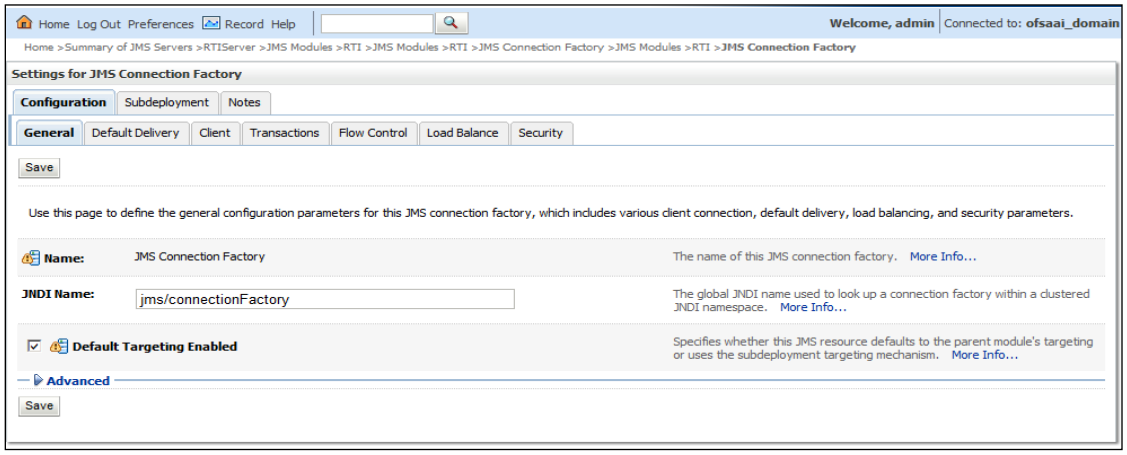
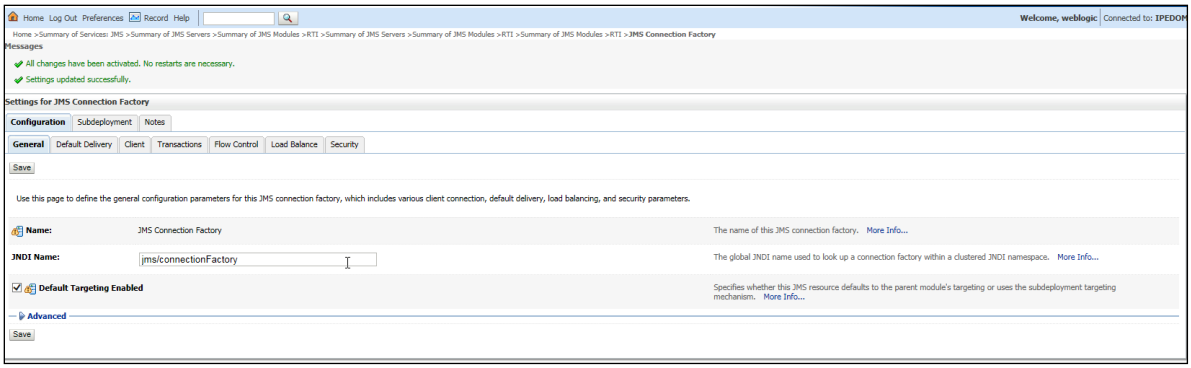


Figure 20: settings for JMS Connection Factory

13. Enter the following details:

Field	Value
Name	The name of the JMS connection factory is displayed as JMS Connection Factory.
JNDI Name	`jms/connectionFactory`
Default Targeting Eanbled	Select the check-box to enable Default Targeting.

14. Click **Save**.
15. The following confirmation message is displayed.
JMS Connection Factory is created successfully.



4.1.6 Creating JMS Topic

This section discusses the following JMS Topics to be created:

- Creating RTI Assessment Response Destination Topic
- Creating Cache Operation Message Destination Topic

4.1.6.1 Creating RTI Assessment Response Destination Topic

To create JMS Topic, follow these steps:

1. In the Domain Structure LHS menu, click **+** to expand **Services**.
2. Click **+** to expand **Messaging**.
3. Click **JMS Modules**. The *JMS Modules* screen is displayed.

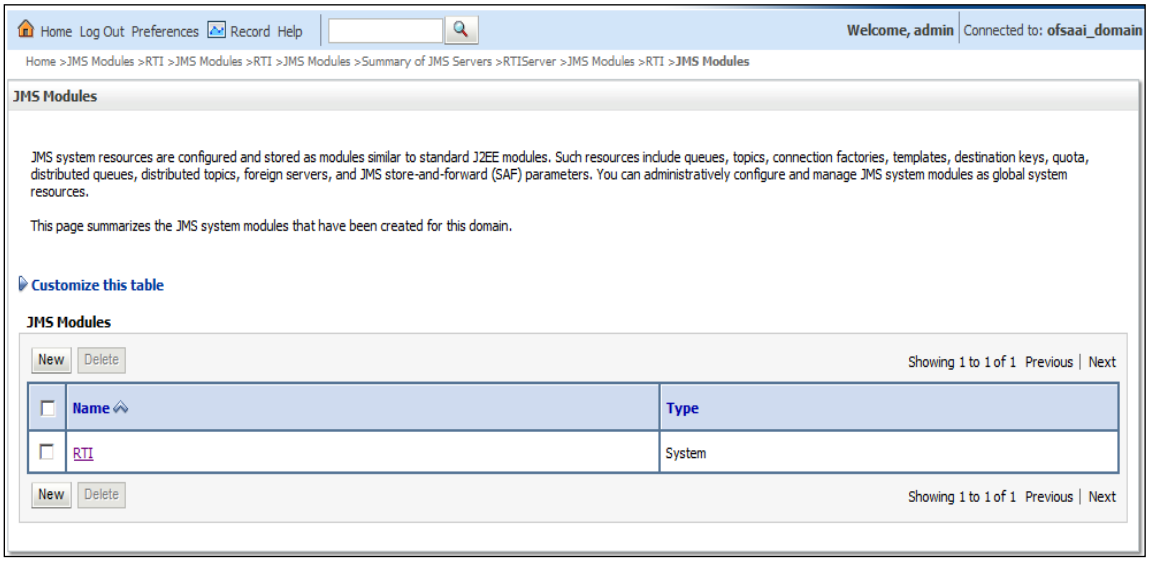


Figure 21: JMS Modules

4. Click **RTI**. The *Settings for RTI* screen is displayed.

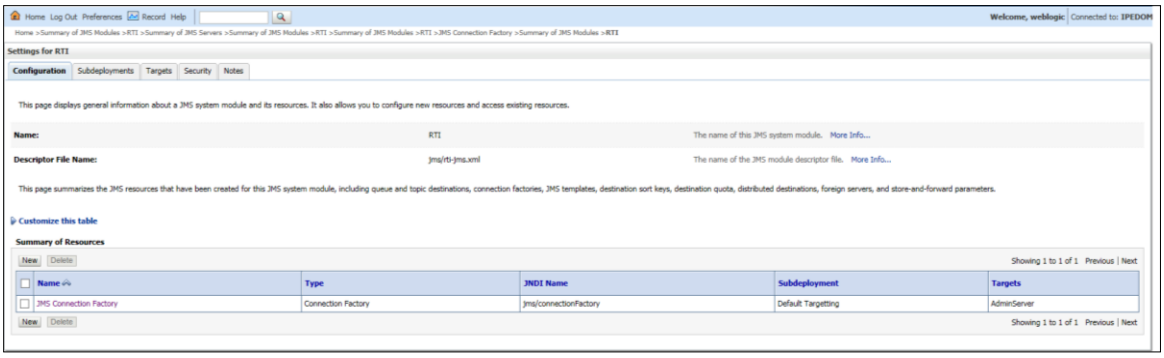


Figure 22: Settings for RTI - JMS topic

5. Click **New**. The *Settings for JMS Connection Factory* screen is displayed.

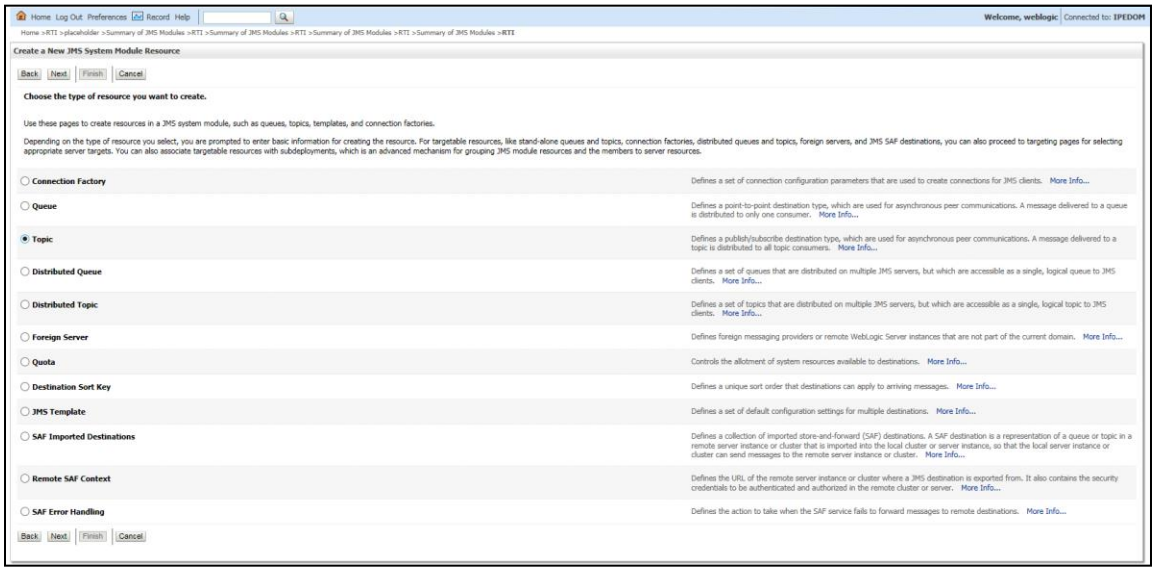


Figure 23: Select type of resource

6. Select **Topic** from the list.
7. Click **Next**. The Create a New JMS System Module Resource screen is displayed.

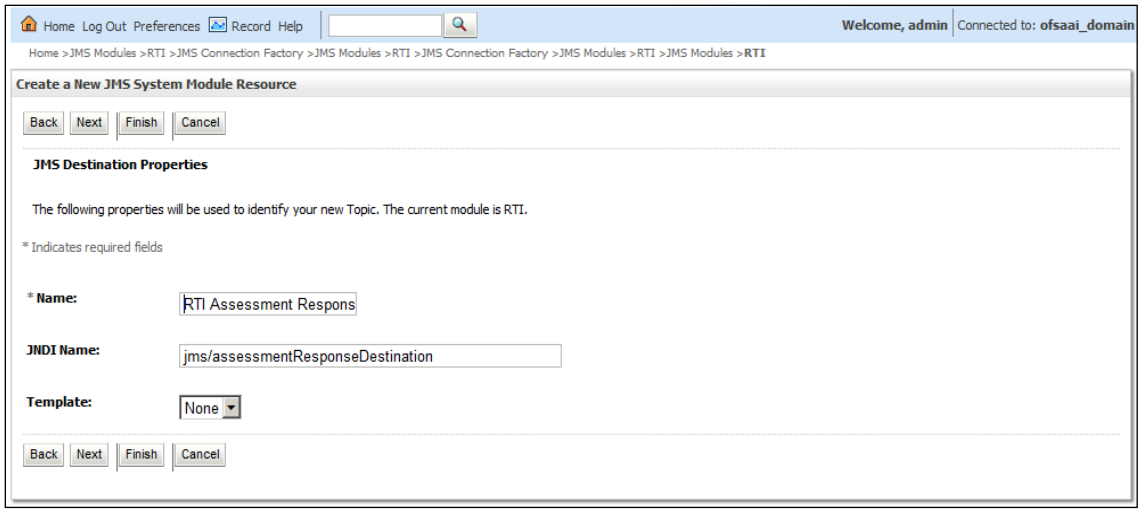


Figure 24: Create a New JMS System Module Resource

8. Enter the following details:

Table 2: JMS Topic - Field Values

Field	Value
Name	RTI Assessment Response Destination Topic
JNDI Name	`jms/assessmentResponseDestination`

9. Click **Next**. The *Create a New JMS System Module Resource* screen is displayed.

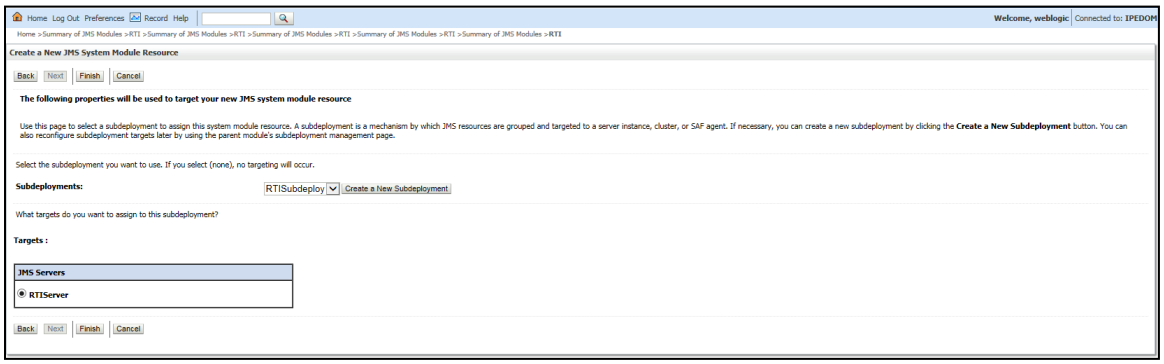


Figure 25: Create a New JMS System Module Resource

10. Select the Subdeployments as **RTISubDeploy**.

11. Select **RTISever**.

12. Click **Finish**.

13. The following confirmation message is displayed.
JMS Topic is created successfully.

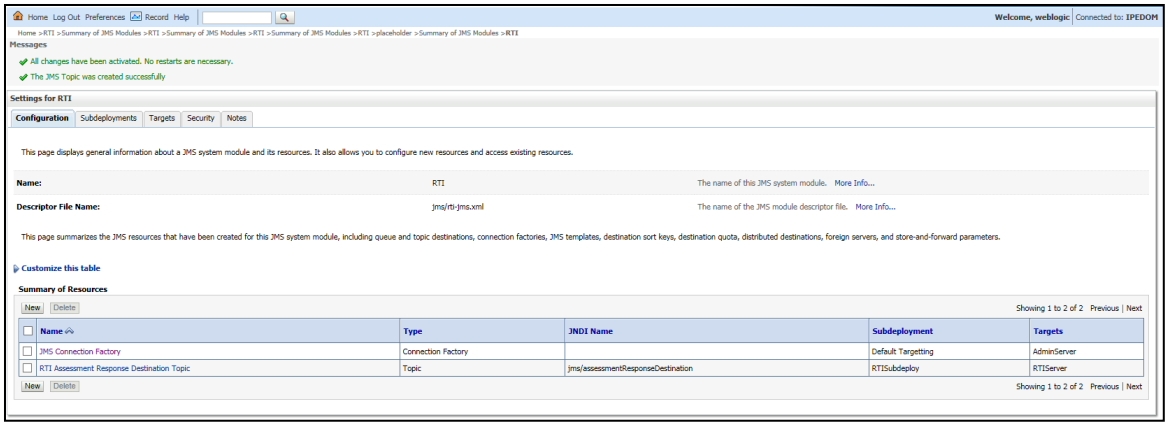


Figure 26: JMS Topic created successfully

4.1.6.2 Creating Cache Operation Message Destination Topic

To create Cache Operation Message Destination Topic, follow these steps:

1. Repeat Steps 1 - 13 from section Creating RTI Assessment Response Destination Topic.
2. Enter the following details:

Table 3: JMS Topic - Field Values

Field	Value
Name	Cache Operation Message Destination Topic
JNDI Name	`jms/cacheOperationMessageDestination`

3. The following confirmation message is displayed.
JMS Topic is created successfully.

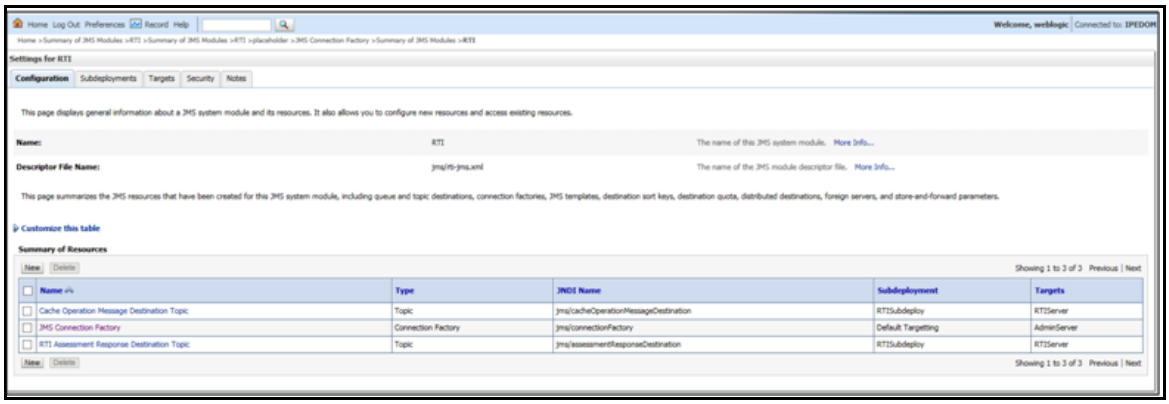


Figure 27: JMS Topic created

4.1.7 Creating JMS Queues

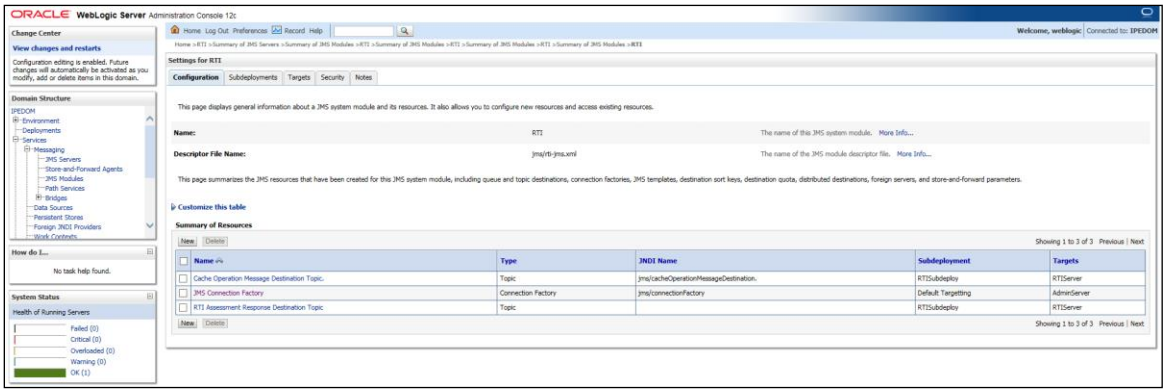
This section discusses the following queues to be created:

- RTI Feedback Queue
- RTI Source Entity Queue
- Wire Transaction Source Entity Queue
- RTI Hold JMS Queue

4.1.7.1 RTI Feedback Queue

To create the RTI Feedback Queue, follow these steps:

1. In the Domain Structure LHS menu, click **+** to expand **Services**.
2. Click **+** to expand **Messaging**.
3. Click JMS Modules.
4. Click **RTI**. The *Settings for RTI* screen is displayed.



5. Click **New**. The Create a New JMS System Module Resource screen is displayed.

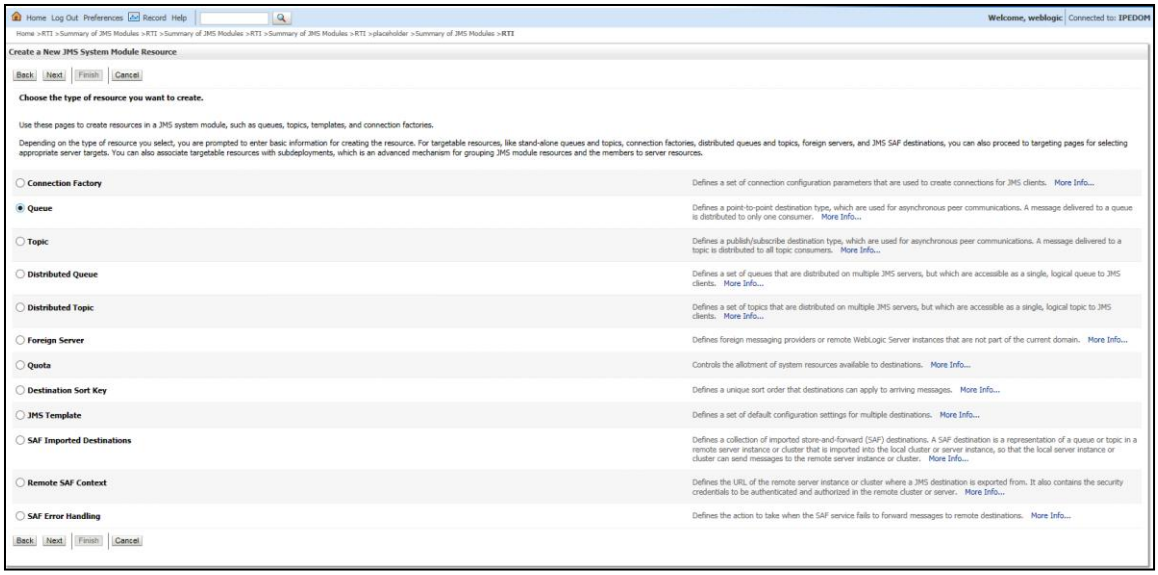


Figure 28: JMS Queue - Create a New JMS system Module

6. Select **Queue** from the list.
7. Click **Next**. The Create a New JMS System Module Resource screen is displayed.

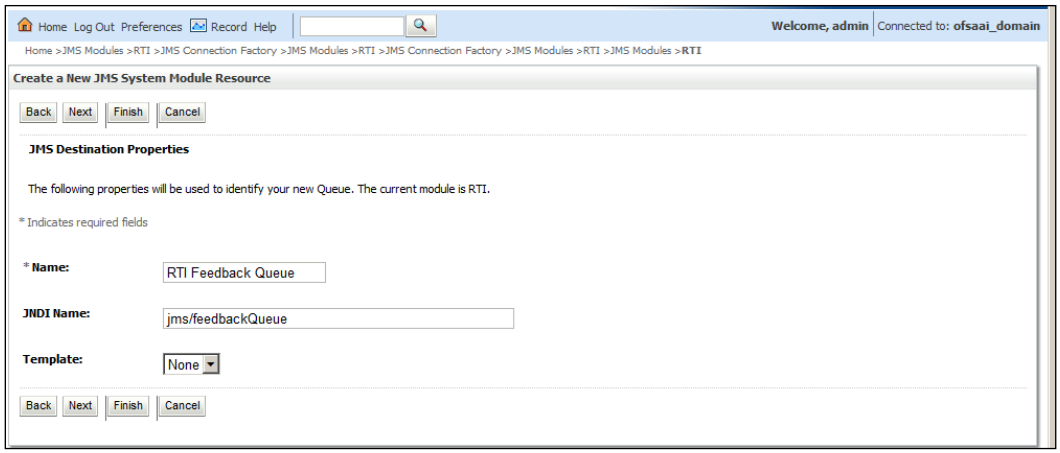


Figure 29: JMS Queue - Create a New JMS system Module

8. Enter the following details:

Table 4: JMS Queue - Field Values

Field	Value
Name	RTI Feedback Queue
JNDI Name	jms/feedbackQueue

9. Click **Next**.
10. Select the Subdeployments as **RTISubDeploy**.

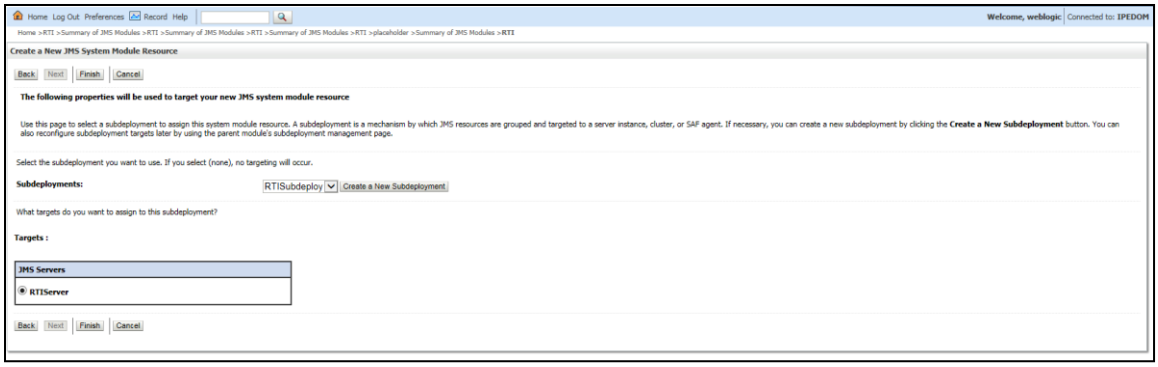


Figure 30: JMS Queue - Create a New JMS System Module Resource

11. Click **Finish**.
12. The following confirmation message is displayed.
    RTI Feedback Queue is created *successfully*.

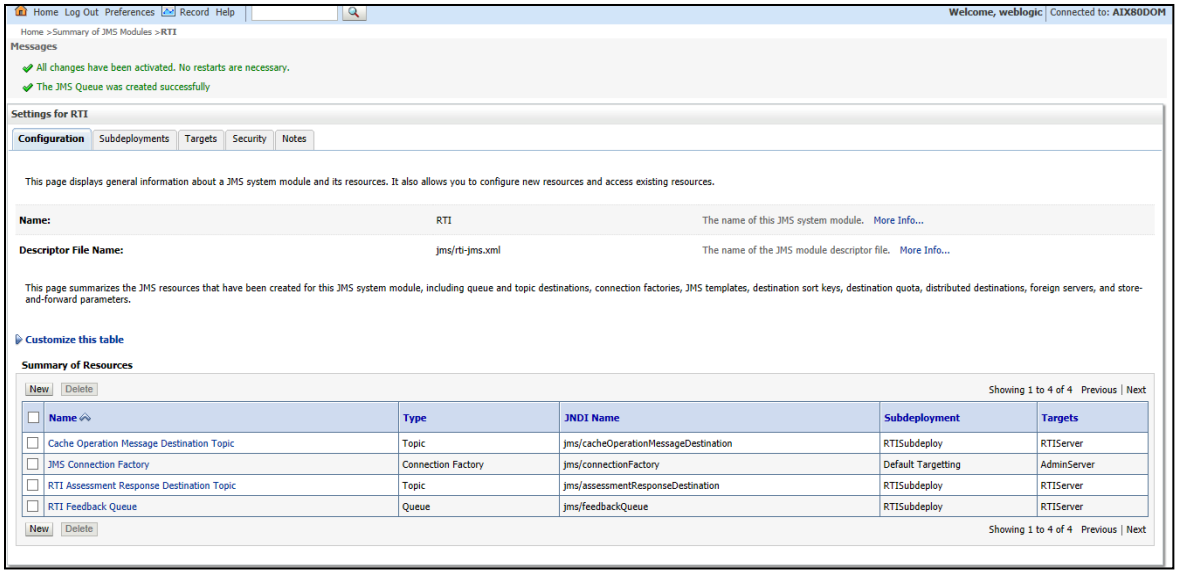


Figure 31: JMS Queue created successfully

4.1.7.2 Creating Remaining JMS Queues

To create the remaining JMS Queues, follow these steps:

1. Repeat Steps 1 - 11 as in section RTI feedback Queue.
2. Enter the values given in the following table.

Table 5: WebLogic JMS Queues - Field Values

Queue Name	Fields		
	Name	JNDI name	Subdeployment
RTI Hold JMS Queue	Enter the name as RTI Hold JMS Queue	Enter the JNDI name as jms/TransactionActionQueue	Select the Subdeployment as RTISubDeploy
RTI Source Entity Queue	Enter the name as RTI Source Entity Queue	Enter the JNDI name as jms/sourceEntityQueue	Select the Subdeployment as RTISubDeploy
Wire Transaction Source Entity Queue	Enter the name as Wire Transaction Source Entity Queue	Enter the JNDI name as jms/wireTrxnQueue	Select the Subdeployment as RTISubDeploy

3. The following confirmation message is displayed.
The JMS Queue was created successfully.

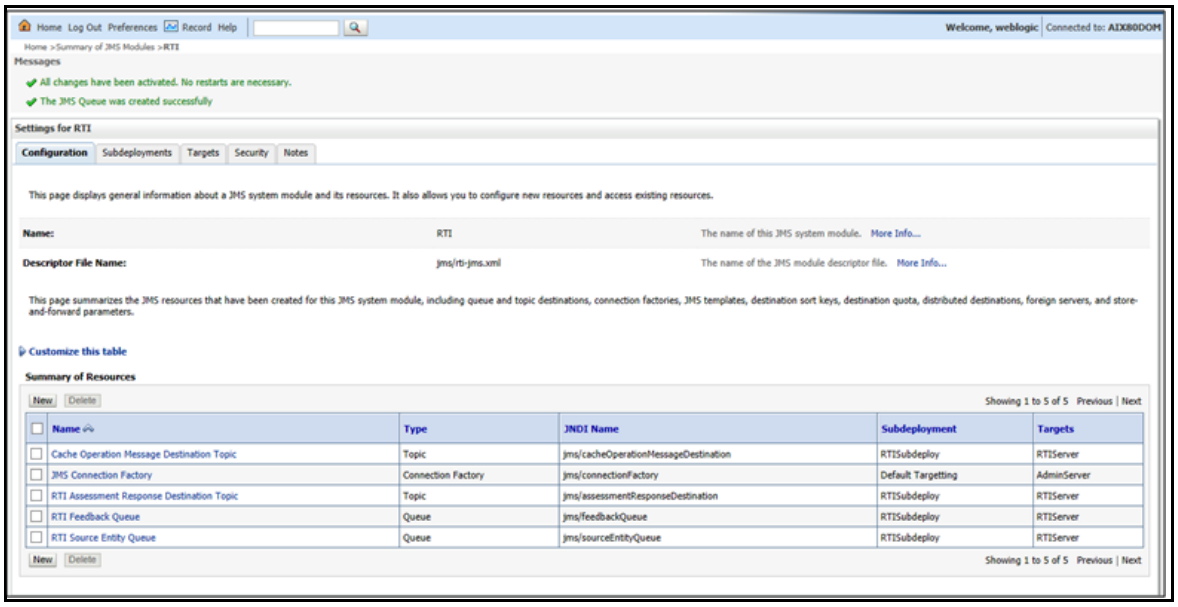


Figure 32: JMS Queues

4.1.8 Restart Weblogic Domain

For more information, refer to the **Start/Stop Infrastructure Services** section in the *Oracle Financial Services Analytical Applications Infrastructure Installation and Configuration Guide* available on the OHC page.

4.2 Configuring IPE in WebSphere

This section explains the WebSphere configuration for IPE and includes the following topics:

- Login to WebSphere
- JMS Providers
- JMS Connection Factories
- JMS Queues
- JMS Topics

4.2.1 Login to WebSphere

To configure IPE on WebSphere follow these steps:

1. Open the following URL in the browser window: http://<ipaddress>:<administrative console port>/ibm/console. (https if SSL is enabled). The Login window is displayed.
2. Login with the Administrator **Username** and **Password**.

4.2.2 Bus Creation

1. Click + to expand **Service Integration** in the LHS menu.
2. Click **Buses**. The Buses page is displayed.

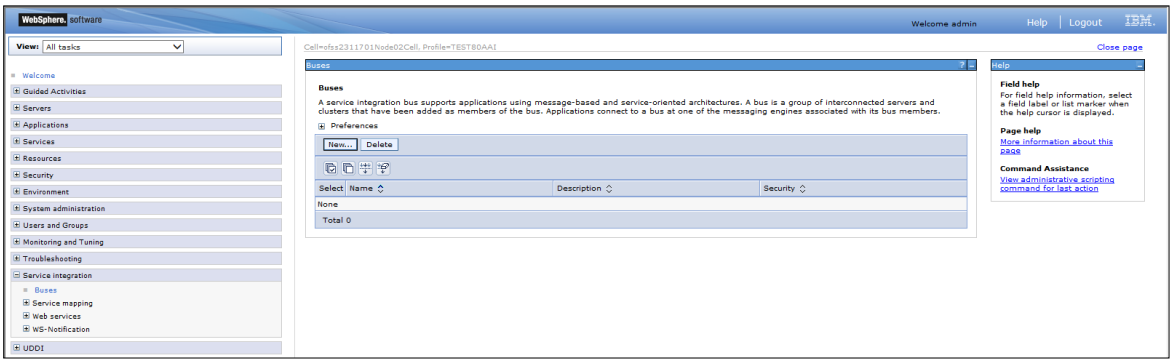


Figure 33: Buses

3. Click **New**. The Create a New Service Integration Bus screen is displayed

Figure 34: Create a New Service Integration Bus

4. Enter the name as **RTIServer**.
5. Un-check Bus security.
6. Click **Next**.

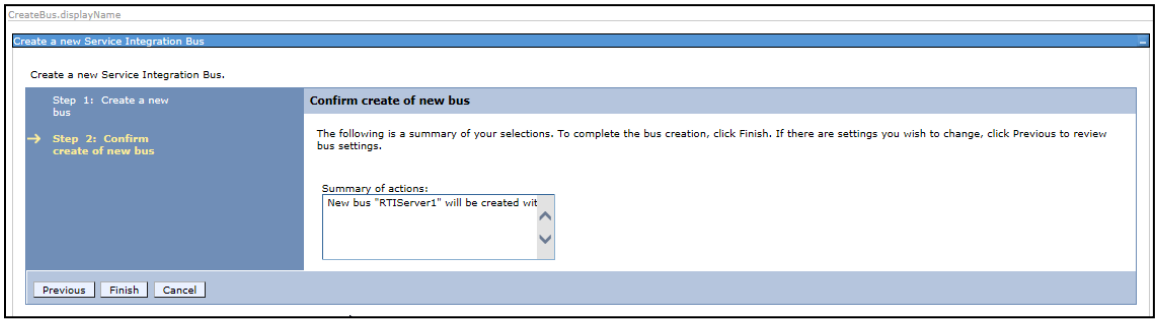


Figure 35: Create a New Service Integration Bus

7. Click **Finish**.
8. Click **Save**.

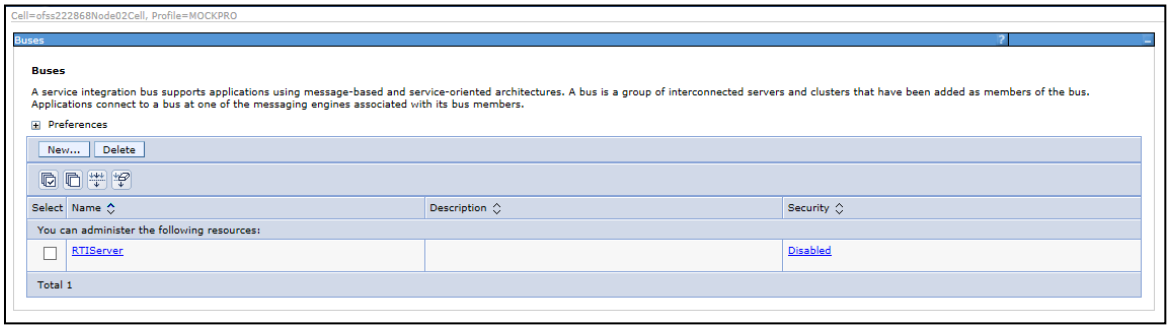


Figure 36: Buses

4.2.3 Bus Member Creation

To create a Bus Member follow these steps:

1. Click + to expand **Service Integration** in the LHS menu.
2. Click **Buses**.
3. Click **RTIServer**. The RTI Server screen is displayed.

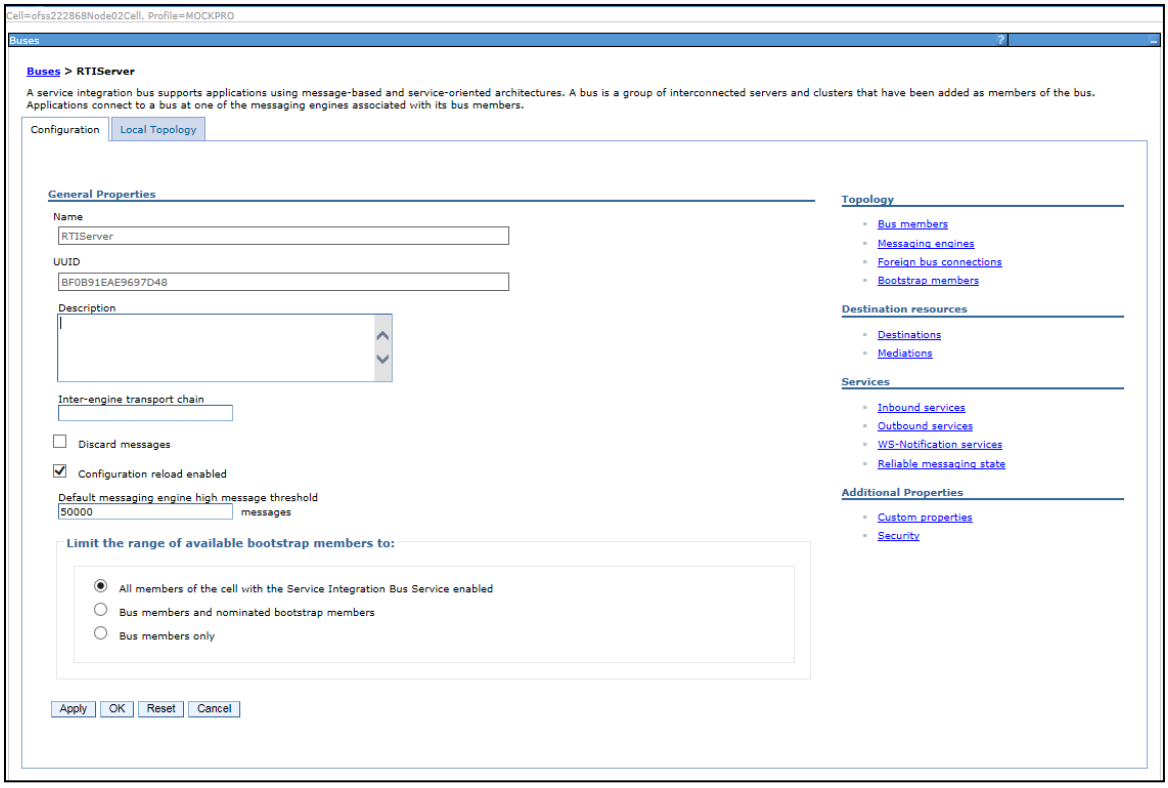


Figure 37: RTI Server

4. In the Topology section, click **Bus members**. The Bus members screen is displayed.

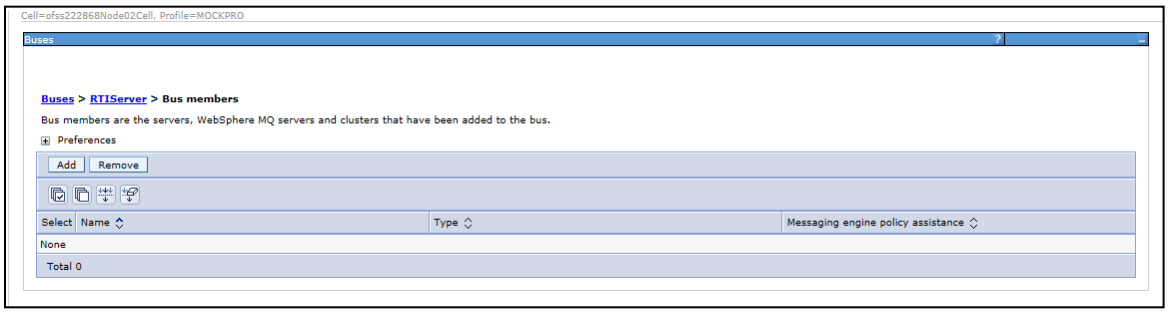


Figure 38: Bus members

5. Click **Add**. The Add a New Bus Member screen is displayed.
6. Select **Server**.

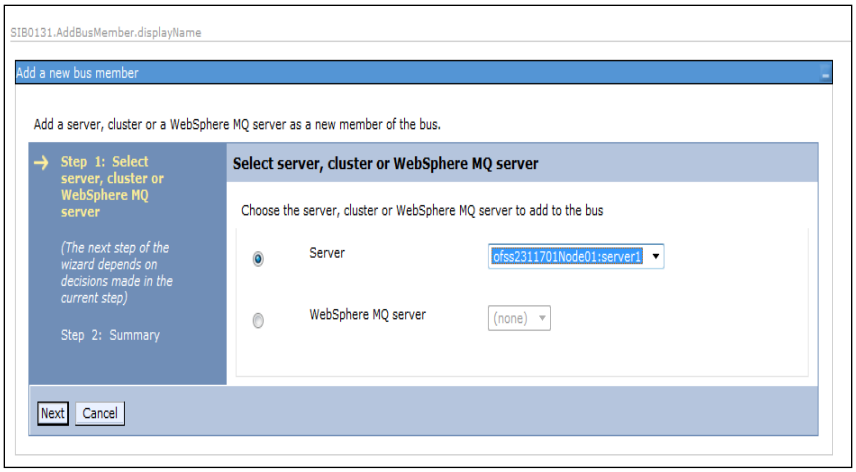
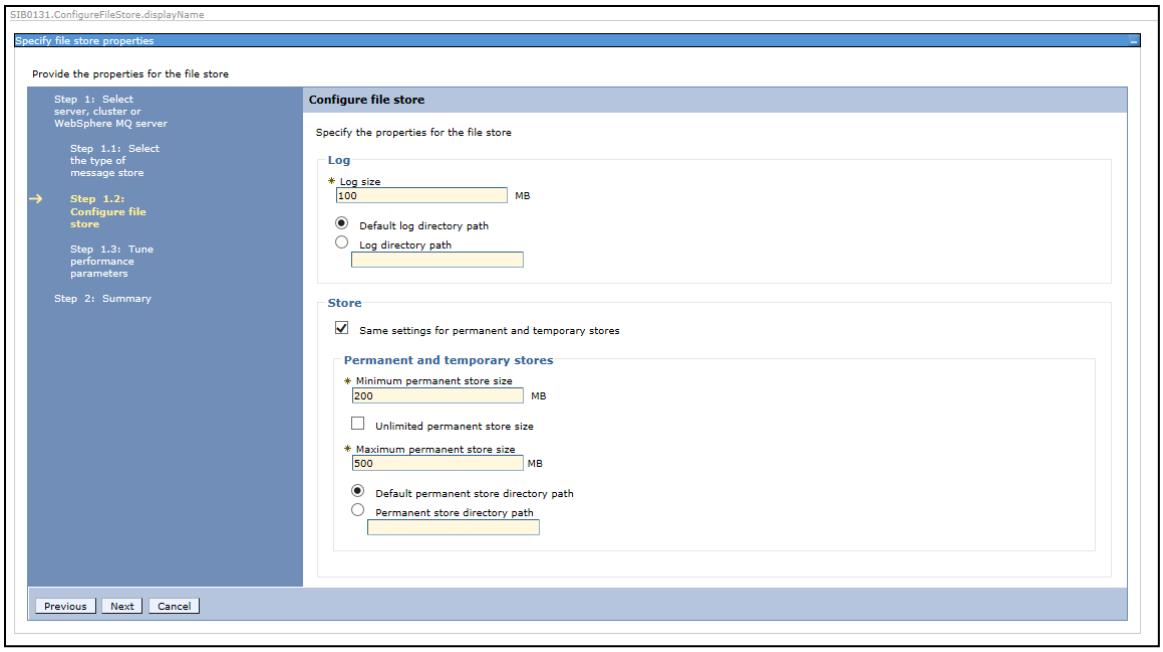


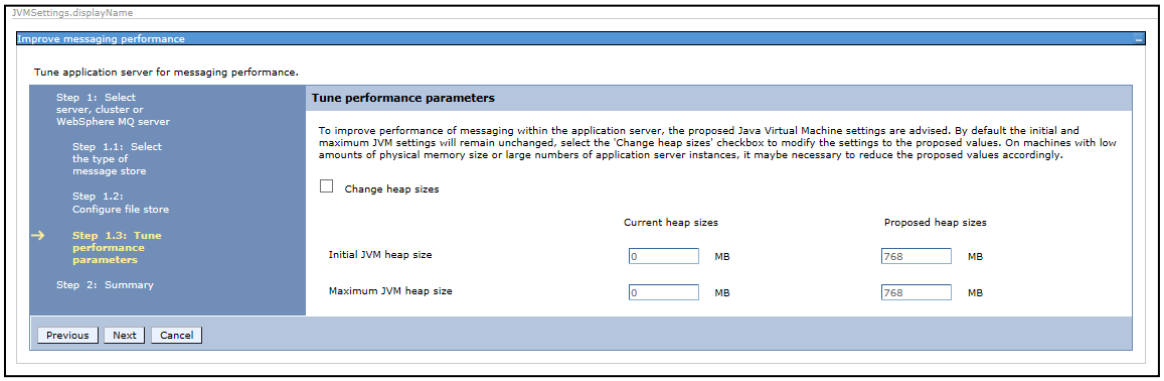
Figure 39: Add a New Bus Member

7. Click **Next**.

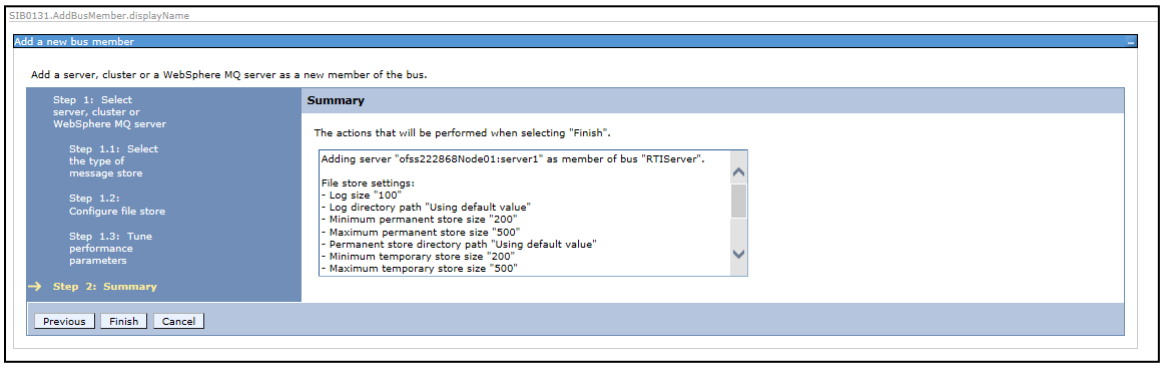
8. Select **File Store**.
9. Click **Next**.



10. Click **Next.**



11. Click **Next.**



12. Click **Finish**. The Buses screen is displayed.

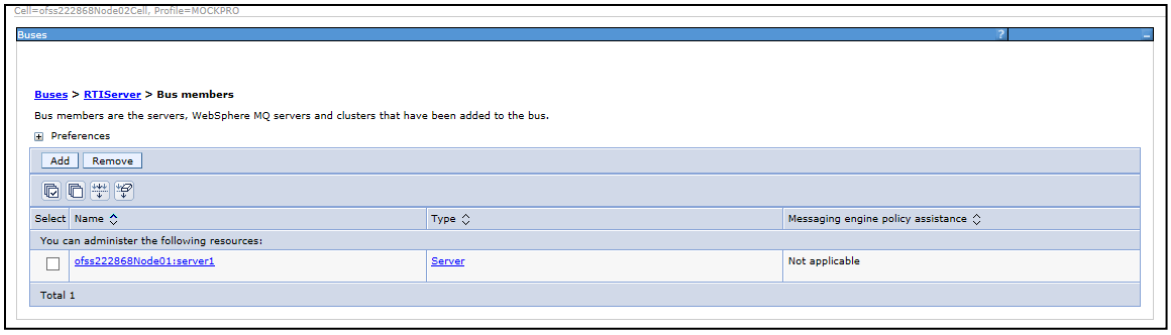


Figure 40: Bus Members created successfully

13. Click **Save**.

4.2.4 Configuring JMS Providers

This section discusses the configuration of JMS providers.

To navigate to JMS Providers sections, follow these steps:

1. Click **+** to expand **Resources**.
2. Click **+** to expand **JMS**.
3. Click **JMS Providers**. The JMS Providers screen is displayed.

4. Select **Cell** as Scope. (for example, Cell=OFSA80Node02Cell)
5. Verify that the Default messaging provider exists.

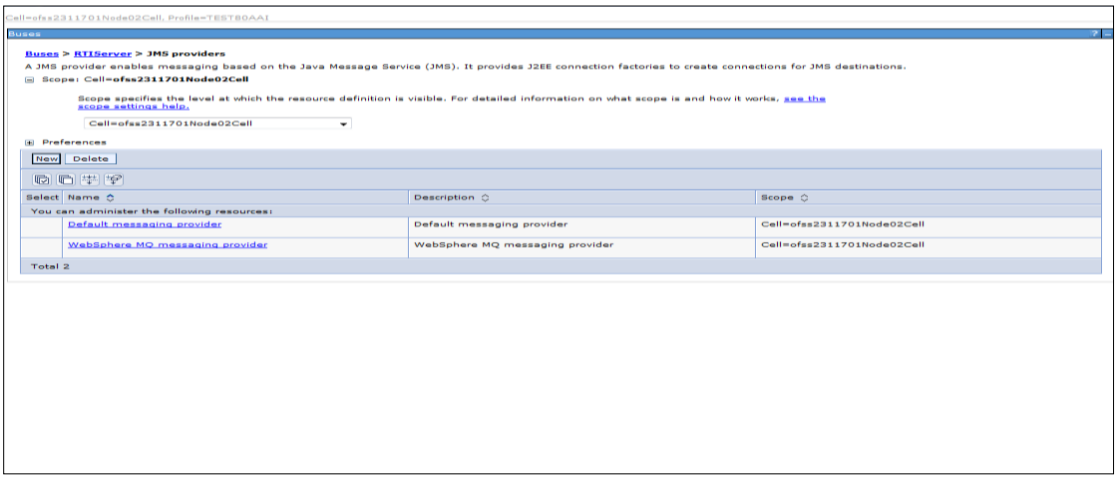


Figure 41: JMS Providers

4.2.5 Configuring JMS Connection Factory

This section explains about configuring JMS Connection Factory.

> **NOTE** For information about the ports used, refer to section Check Ports in WebSphere.

To configure JMS Connection Factory, follow these steps:

1. Click **+** to expand **Resources**.
2. Click **+** to expand **JMS**.
3. Click **Connection Factories**. The Connection Factories screen is displayed.

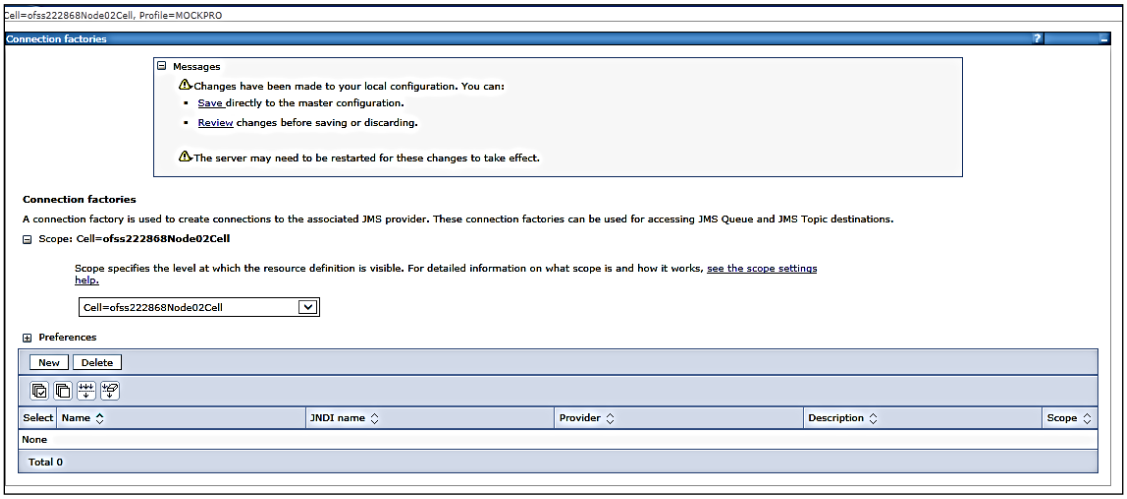


Figure 42: Connection Factories

4. Select the Scope as **Cell**. (for example, Cell=OFSA80Node02Cell)
5. Click **New**.
6. Select Default Messaging Provider option.
7. Click **OK**. The *JMS Connection Factory* screen is displayed.

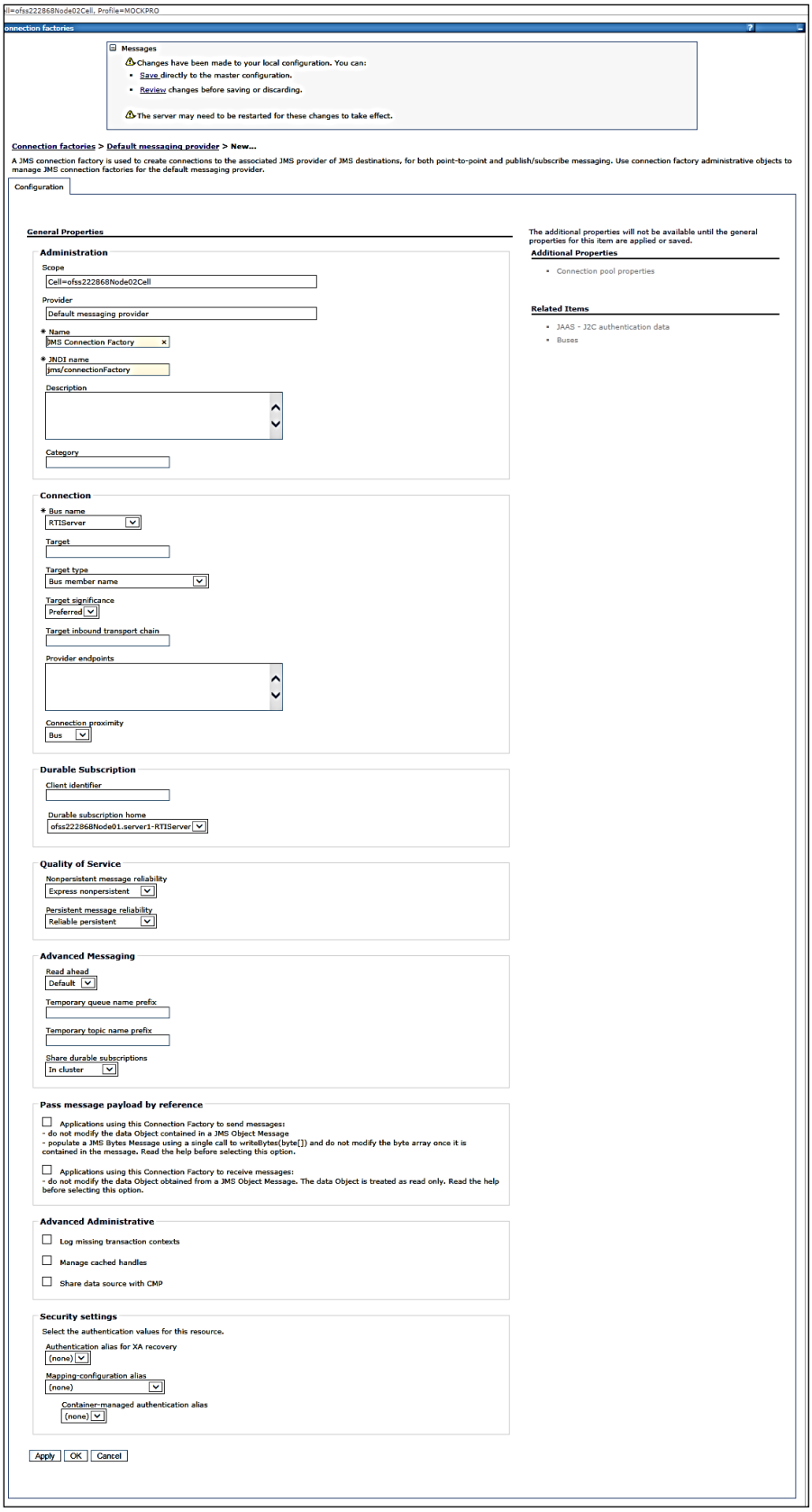

ll=ofss222868Node02Cell, Profile=MOCKPRO
onnection factories
Messages
Changes have been made to your local configuration. You can:
Save directly to the master configuration.
Review changes before saving or discarding.
The server may need to be restarted for these changes to take effect.
Connection factories > Default messaging provider > New...
A JMS connection factory is used to create connections to the associated JMS provider of JMS destinations, for both point-to-point and publish/subscribe messaging. Use connection factory administrative objects to manage JMS connection factories for the default messaging provider.
Configuration
General Properties
Administration
Scope
Cell=ofss222868Node02Cell
Provider
Default messaging provider
* Name
JMS Connection Factory
* JNDI name
jms/connectionFactory
Description
Category
Connection
* Bus name
RTIServer
Target
Target type
Bus member name
Target significance
Preferred
Target inbound transport chain
Provider endpoints
Connection proximity
Bus
Durable Subscription
Client identifier
Durable subscription home
ofss222868Node01.server1-RTIServer
Quality of Service
Nonpersistent message reliability
Express nonpersistent
Persistent message reliability
Reliable persistent
Advanced Messaging
Read ahead
Default
Temporary queue name prefix
Temporary topic name prefix
Share durable subscriptions
In cluster
Pass message payload by reference
Applications using this Connection Factory to send messages:
- do not modify the data Object contained in a JMS Object Message
- populate a JMS Bytes Message using a single call to writeBytes(byte[]) and do not modify the byte array once it is contained in the message. Read the help before selecting this option.
Applications using this Connection Factory to receive messages:
- do not modify the data Object obtained from a JMS Object Message. The data Object is treated as read only. Read the help before selecting this option.
Advanced Administrative
Log missing transaction contexts
Manage cached handles
Share data source with CMP
Security settings
Select the authentication values for this resource.
Authentication alias for XA recovery
(none)
Mapping-configuration alias
(none)
Container-managed authentication alias
(none)
Apply
OK
Cancel
The additional properties will not be available until the general properties for this item are applied or saved.
Additional Properties
Connection pool properties
Related Items
JAAS - J2C authentication data
Buses

Figure 43: JMS Connection Factory

8. Enter the following details:

Table 6: JMS Connection Factory: Field Values

Field	Value	Description
Name	JMS Connection Factory	Enter the name of JMS Connection Factory
JNDI name	jms/connectionFactory	Enter the JNDI name for the JMS connection factory
Bus Name	RTIServer	Select the bus name.
Target Inbound Transport Chain	<Transport Chain Name>	Enter the transport chain name. Refer Appendix C for Transport chain name. For example: InboundBasicMessaging
Provider endpoints	<HOSTNAME> : <SIB_ENDPOINT_ADDRESS port>: <Transport Chain Name>	Enter the transport chain name. Refer Appendix C for Provider endpoints. For example: ofss222868.in.oracle.com:7280:InboundBasicMessaging

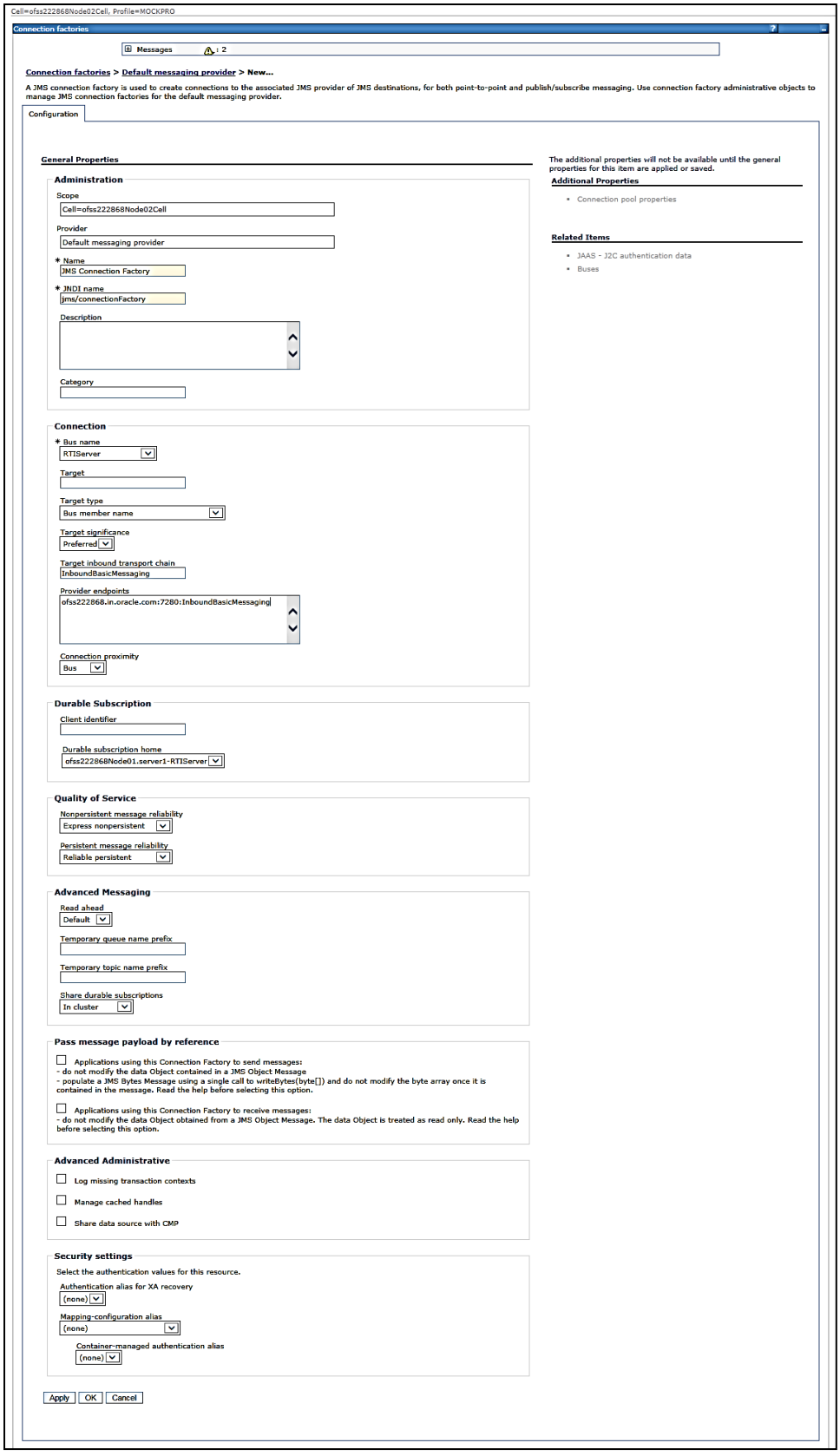

Cell=ofss222868Node02Cell, Profile=MOCKPRO
Connection factories
Messages : 2
Connection factories > Default messaging provider > New...
A JMS connection factory is used to create connections to the associated JMS provider of JMS destinations, for both point-to-point and publish/subscribe messaging. Use connection factory administrative objects to manage JMS connection factories for the default messaging provider.
Configuration
General Properties
Administration
Scope
Cell=ofss222868Node02Cell
Provider
Default messaging provider
* Name
JMS Connection Factory
* JNDI name
jms/connectionFactory
Description
Category
Connection
* Bus name
RTIServer
Target
Target type
Bus member name
Target significance
Preferred
Target inbound transport chain
InboundBasicMessaging
Provider endpoints
ofss222868.in.oracle.com:7280:InboundBasicMessaging
Connection proximity
Bus
Durable Subscription
Client identifier
Durable subscription home
ofss222868Node01.server1-RTIServer
Quality of Service
Nonpersistent message reliability
Express nonpersistent
Persistent message reliability
Reliable persistent
Advanced Messaging
Read ahead
Default
Temporary queue name prefix
Temporary topic name prefix
Share durable subscriptions
In cluster
Pass message payload by reference
Applications using this Connection Factory to send messages:
- do not modify the data Object contained in a JMS Object Message
- populate a JMS Bytes Message using a single call to writeBytes(byte[]) and do not modify the byte array once it is contained in the message. Read the help before selecting this option.
Applications using this Connection Factory to receive messages:
- do not modify the data Object obtained from a JMS Object Message. The data Object is treated as read only. Read the help before selecting this option.
Advanced Administrative
Log missing transaction contexts
Manage cached handles
Share data source with CMP
Security settings
Select the authentication values for this resource.
Authentication alias for XA recovery
(none)
Mapping-configuration alias
(none)
Container-managed authentication alias
(none)
Apply
OK
Cancel
The additional properties will not be available until the general properties for this item are applied or saved.
Additional Properties
Connection pool properties
Related Items
JAAS - J2C authentication data
Buses

Figure 44: JMS Connection Factory – Not default port

9. Click **Apply** and save the details.

4.2.6 Configuring JMS Queues

This section discusses the following JMS Queues which are to be created:

- RTI Source Entity Queue
- RTI Hold JMS Queue
- RTI Feedback Queue
- Wire Transaction Source Entity Queue

4.2.6.1 Configuring RTI Source Entity Queue

To create RTI Source Entity Queue, follow these steps:

1. Click **+** to expand **Resources** in the LHS menu.
2. Click **+** to expand **JMS**.
3. Click **Queues**.

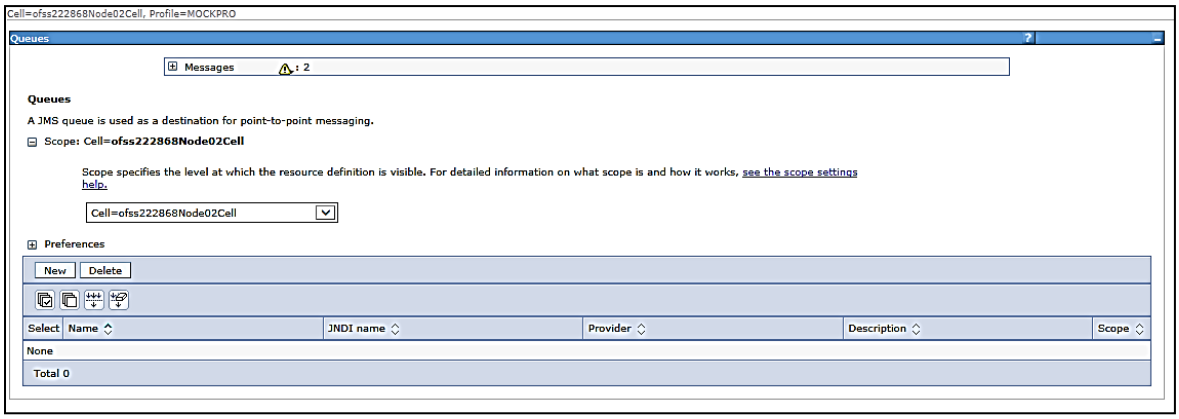


Figure 45: Queues

4. Select Scope as **Cell**. (For example, cell=OFSA80Node02Cell).
5. Click **New**. The *Select JMS resource provider* screen is displayed.

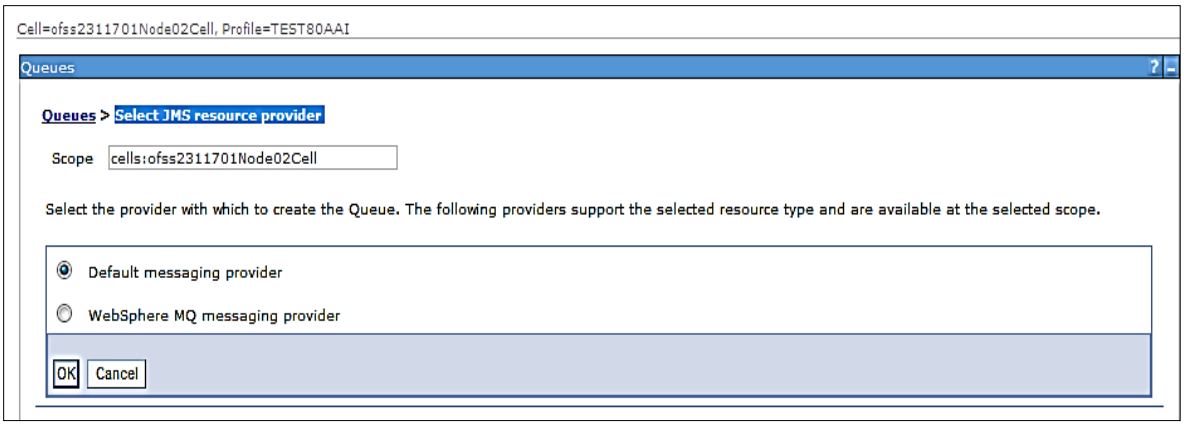


Figure 46: Select JMS resource provider

6. Select Default Messaging Provider.
7. Click **OK**. The General Properties section is displayed.

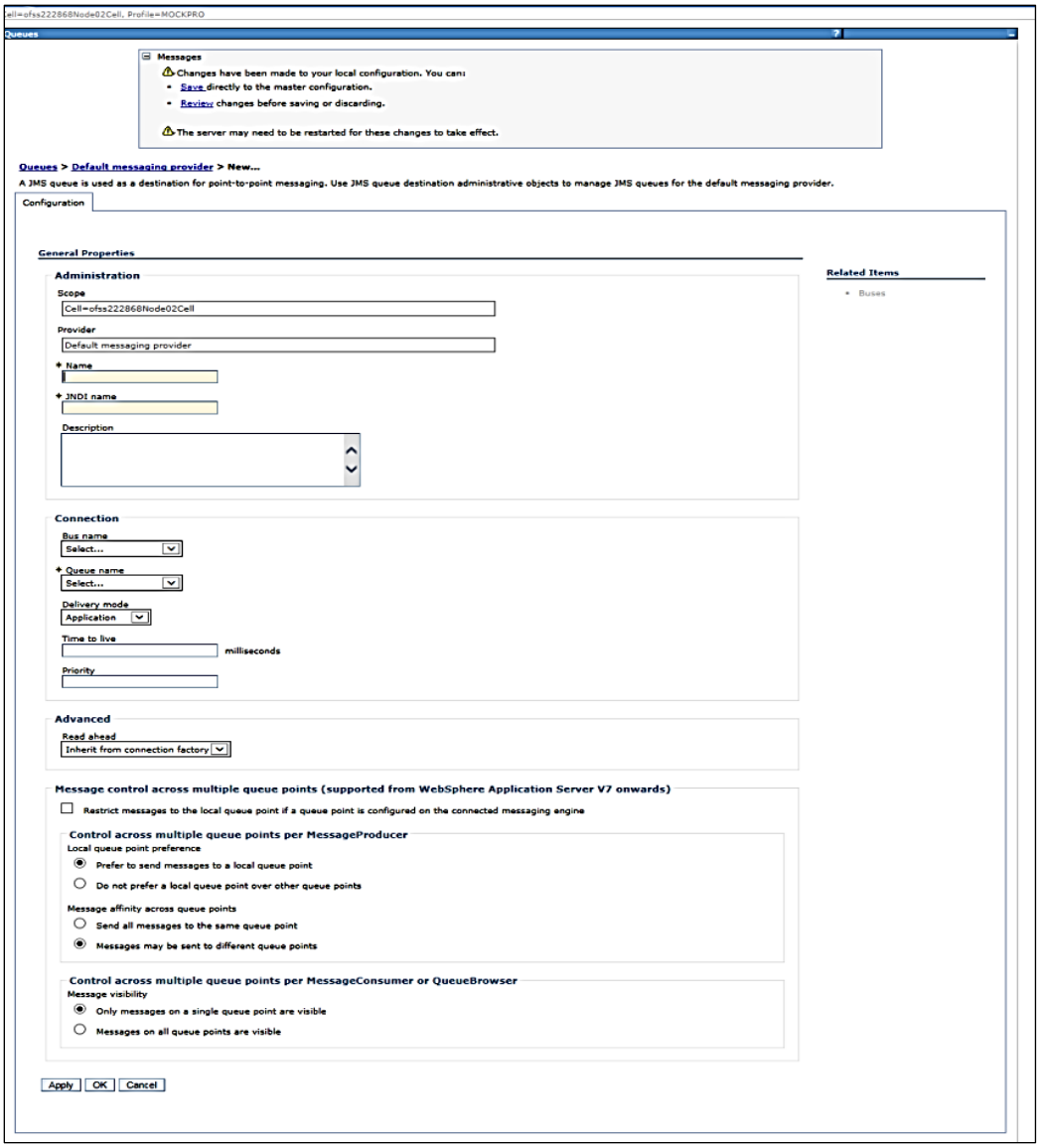


Figure 47: General Properties section

8. Enter the following details:

Table 7: JMS Queues General Properties - Field Values

Field	Value
Name	RTI Source Entity Queue
JNDI Name	jms/sourceEntityQueue
Bus Name	RTIServer

9. Select Create Service Integration Bus destination from Queue Name.

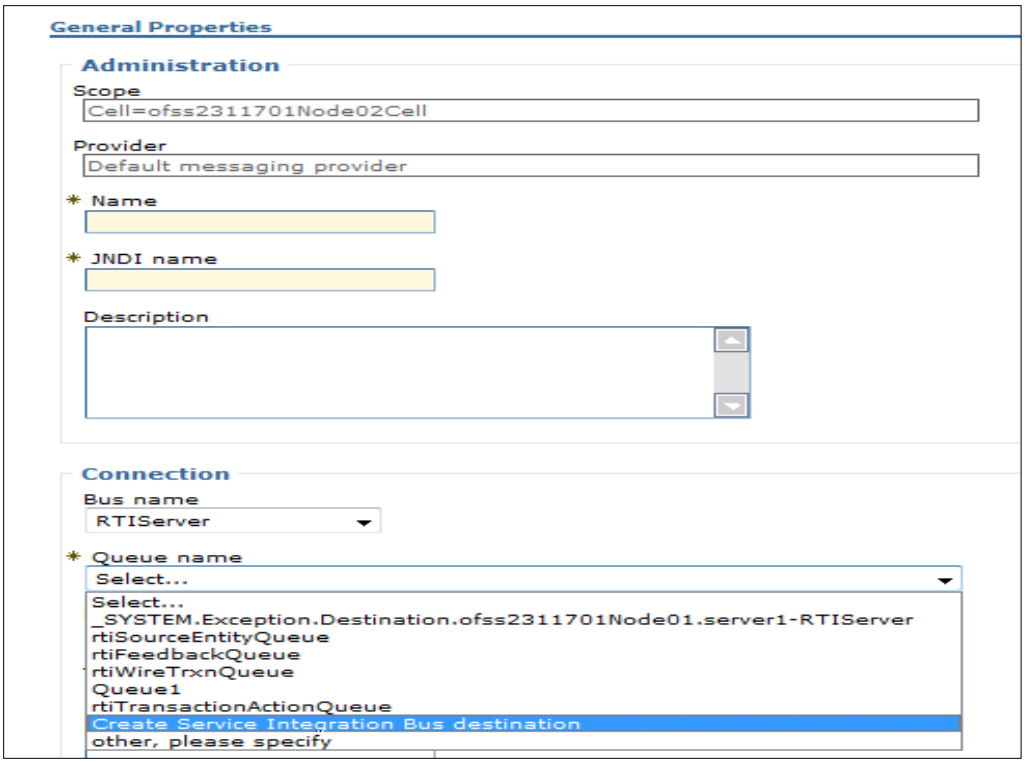


Figure 48: Queue Name

10. The *Set queue attributes* screen is displayed.

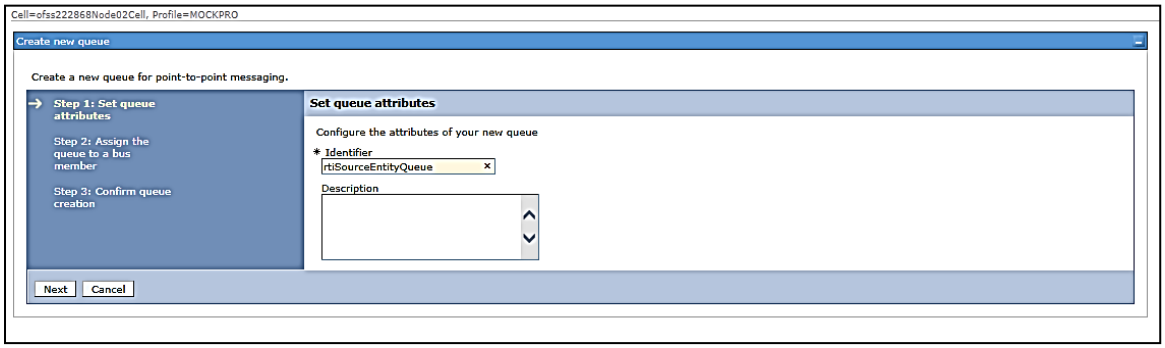


Figure 49: Set queue attributes

11. Enter the Identifier as **rtiSourceEntityQueue**.

12. Click **Next**.

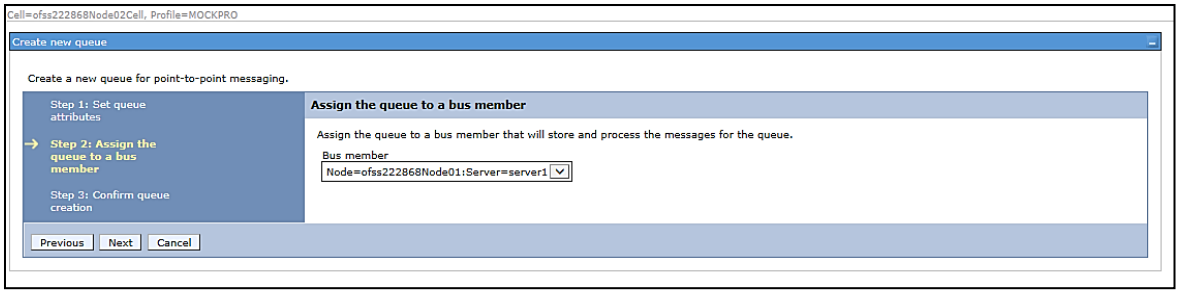


Figure 50: Assign the queue to a bus member

13. Click **Next**.

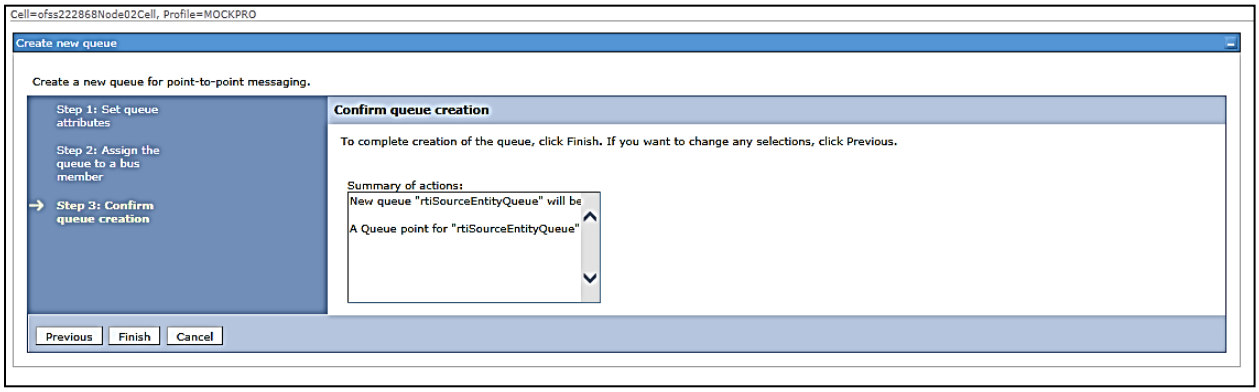


Figure 51: Confirm queue creation

14. Click **Finish**. The Configuration screen is displayed.

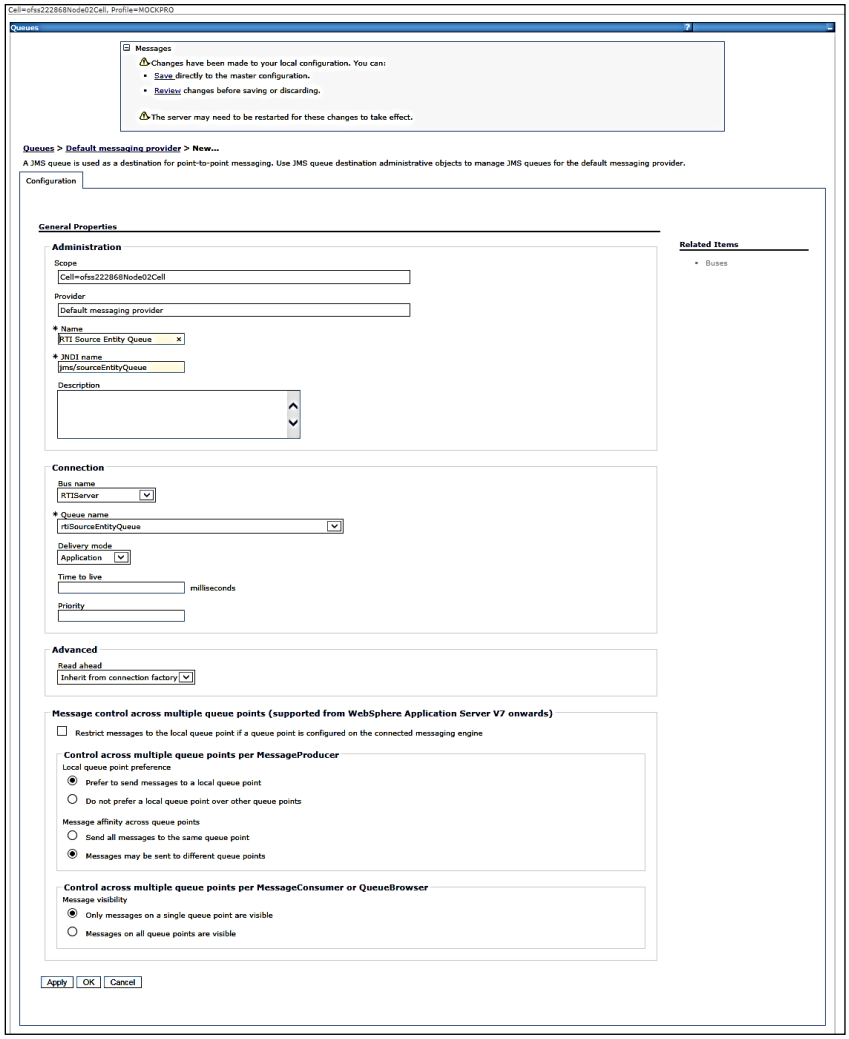


Figure 52: Configuration

15. Click **Apply** and save the details.

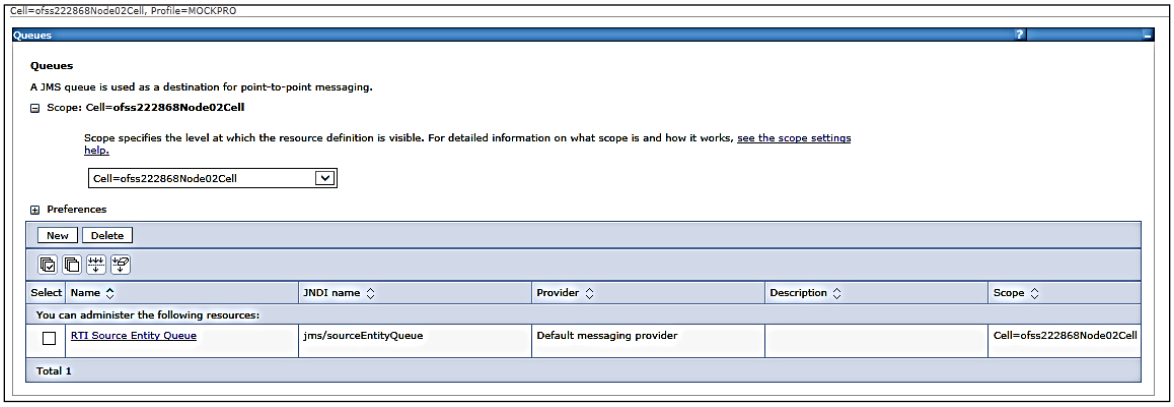


Figure 53: Queues

4.2.6.2 Creating remaining JMS Queues

Similarly, to create the remaining queues, follow these steps:

1. Repeat Steps 1 - 15 from section RTI Source Entity Queue.
2. Enter the following details:

Table 8: WebSphere JMS Queues - Field Values

Queue Name	Fields			
	Name	**JNDI name**	**Bus name**	**Queue Identifier**
RTI Hold JMS Queue	Enter the name as RTI Hold JMS Queue	Enter the JNDI name as jms/TransactionActionQueue	Select the Bus name as RTIServer	Enter the Queue as rtiTransactionActionQueue
RTI Feedback Queue	Enter the name as RTI Feedback Queue	Enter the JNDI name as jms/feedbackQueue	Select the Bus name as RTIServer	Enter the Queue as rtiFeedbackQueue
Wire Transaction Source Entity Queue	Enter the name as Wire Transaction Source Entity	Enter the JNDI name as jms/wireTrxnQueue	Select the Bus name as RTIServer	Enter the Queue as rtiWireTrxnQueue

3. The JMS Queues are created successfully.

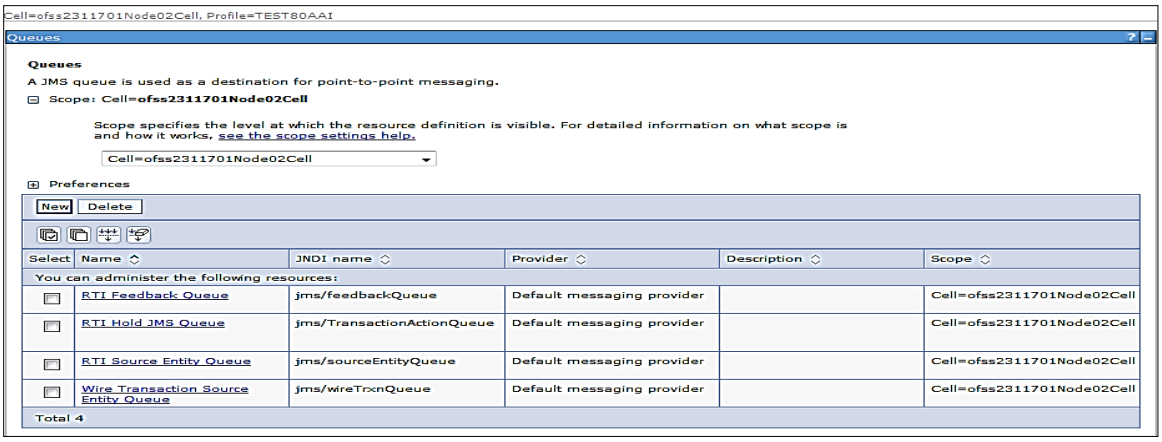


Figure 54: JMS Queue

4.2.7 Configuring JMS Topics

This section discusses the following JMS Topics which are to be created:

- Creating RTI Cache Operation Message Destination Topic
- Creating RTI Assessment Response Destination Topic

4.2.7.1 Creating RTI Cache Operation Message Destination Topic

To create JMS topics, follow these steps:

1. Click **+** to expand **Resources** in the LHS menu.
2. Click **+** to expand **JMS**.
3. Click **Topics**.
4. Select Cell as Scope (for example Cell=OFSA80Node02Cell)

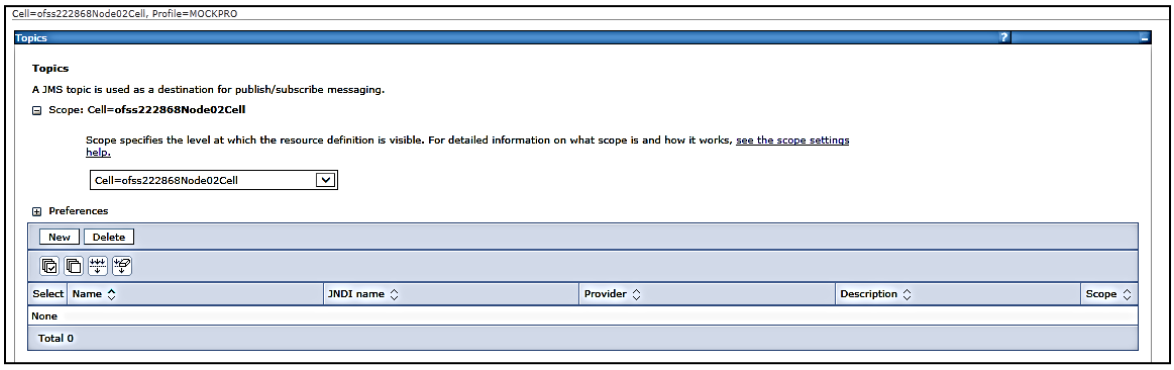


Figure 55: WebSphere - JMS Topics

5. Click **New**. The *Select JMS resource provider* screen is displayed.

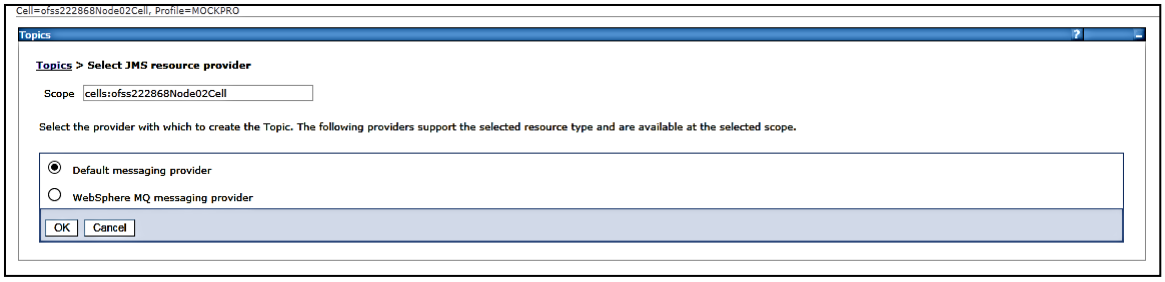


Figure 56: Select JMS resource provider

6. Select Default messaging provider.
7. Click **OK**. The *Configuration* screen is displayed.

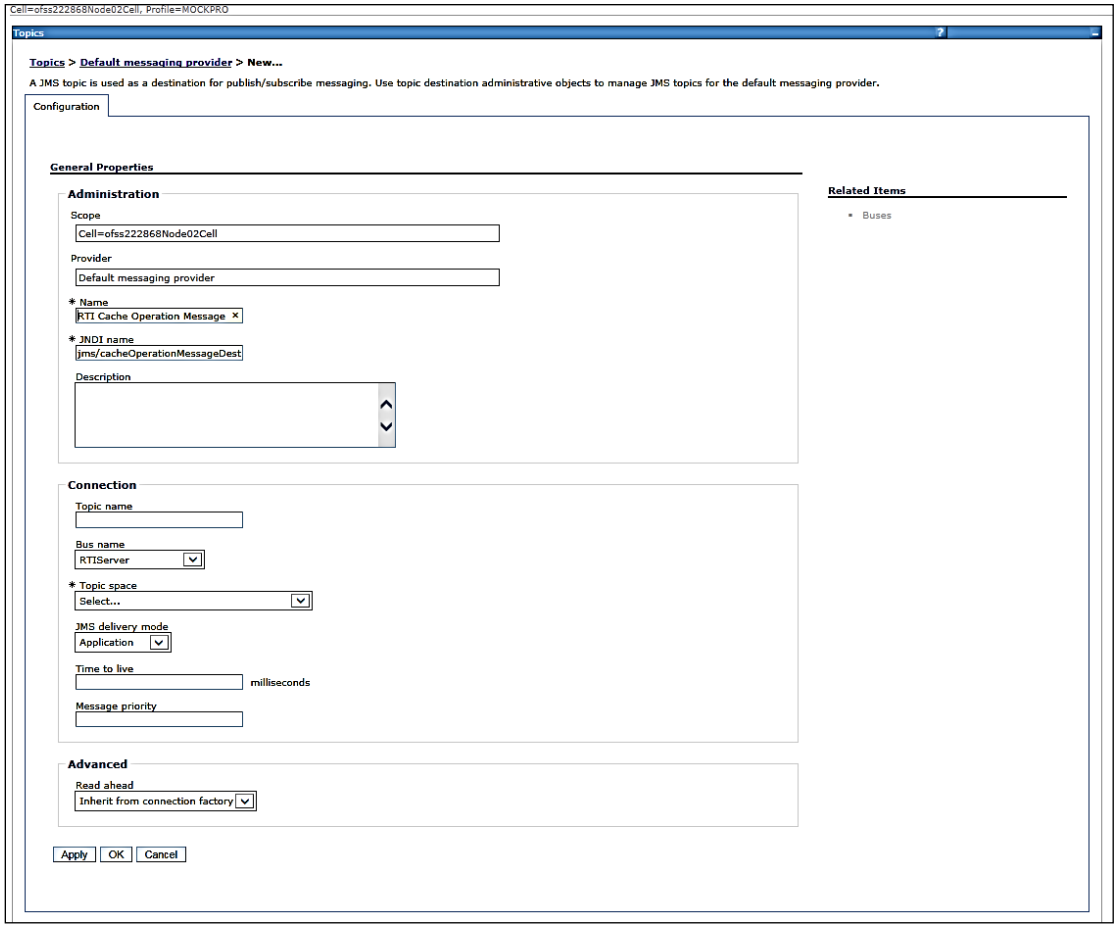


Figure 57: Configuration screen

8. Enter the following details:

Table 9: JMS Topic General Properties - Field Values

Field	Value
Name	RTI Cache Operation Message Destination Topic

Field	Value
JNDI Name	jms/cacheOperationMessageDestination
Bus Name	RTIServer

9. Select Create Service Integration Bus Destination from Topic space.
10. The *Create new topic space* screen is displayed.

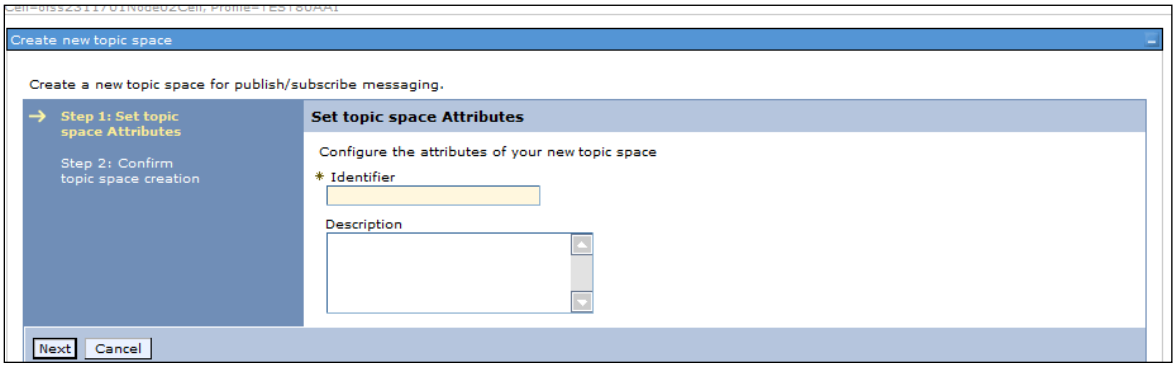


Figure 58: Create new topic space

11. Enter the Identifier as rtiCacheOperationTopic.
12. Click **Next**.

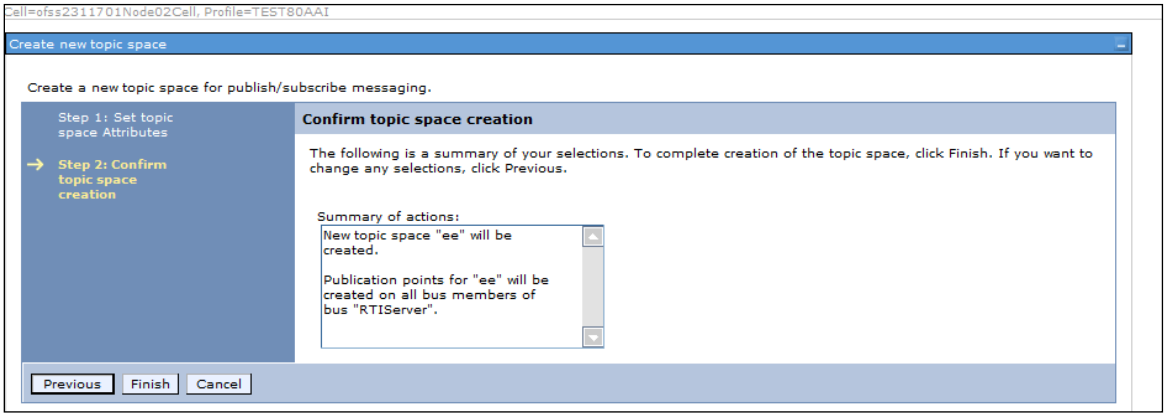


Figure 59: Confirm topic space creation

13. Click **Finish**. The *Configuration* screen is displayed.
14. Click **Apply** and save details.

4.2.7.2 Creating RTI Assessment Response Destination Topic

To create an RTI Assessment Response Destination Topic, follow these steps:

1. Repeat Steps 1-14 from section RTI Cache Operation Message Destination Topic.
2. Enter the following details:

Field	Value
Name	RTI Assessment Response Destination Topic
JNDI name	jms/assessmentResponseDestination
Bus name	RTIServer
Topic Space Identifier	rtiAssessmentResponseDestinationTopic

4.2.8 RMI/IIOP Authentication Settings

This section describes the steps for authentication settings. For security setting, follow these steps:

1. Click **+** to expand **Security** in the LHS menu.
2. Click **+** to expand **Global Security**.
3. Click + to expand **RMI/IIOP security** under **Authentication** section.

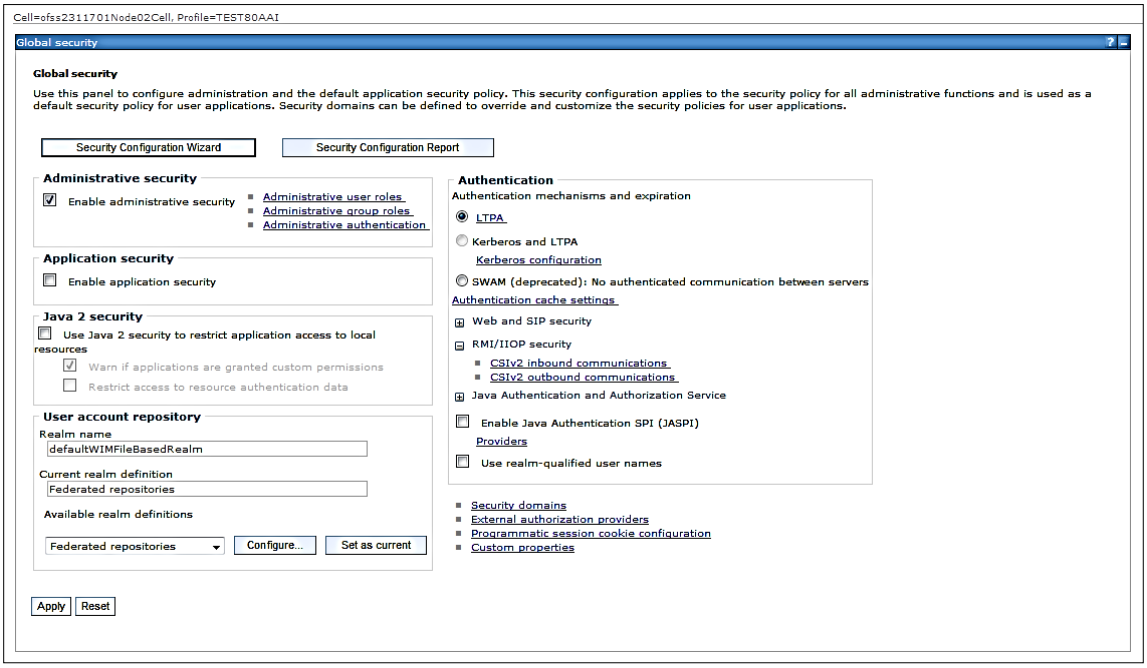


Figure 60: Global Security screen

4. Click CSIv2 inbound communications/CSIv2 outbound communications.
5. Select the following values:

Table 10: RMI/IIOP authentication Settings

RMI/IOP Security	Client certificate authentication	Transport
CSIv2 inbound communications	Supported	SSL-supported
CSIv2 outbound communications	Supported	SSL-supported

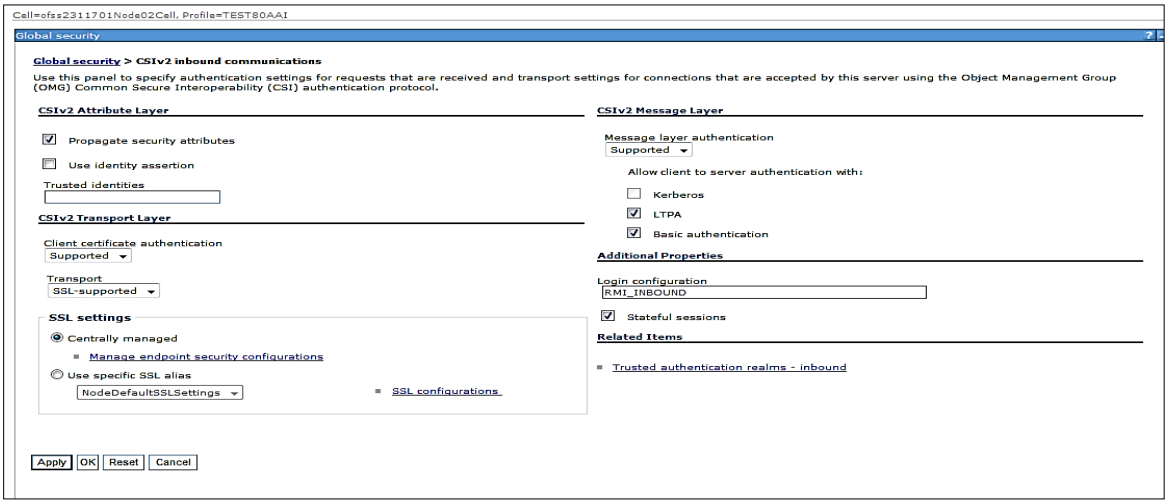


Figure 61: CSIv2 inbound communications

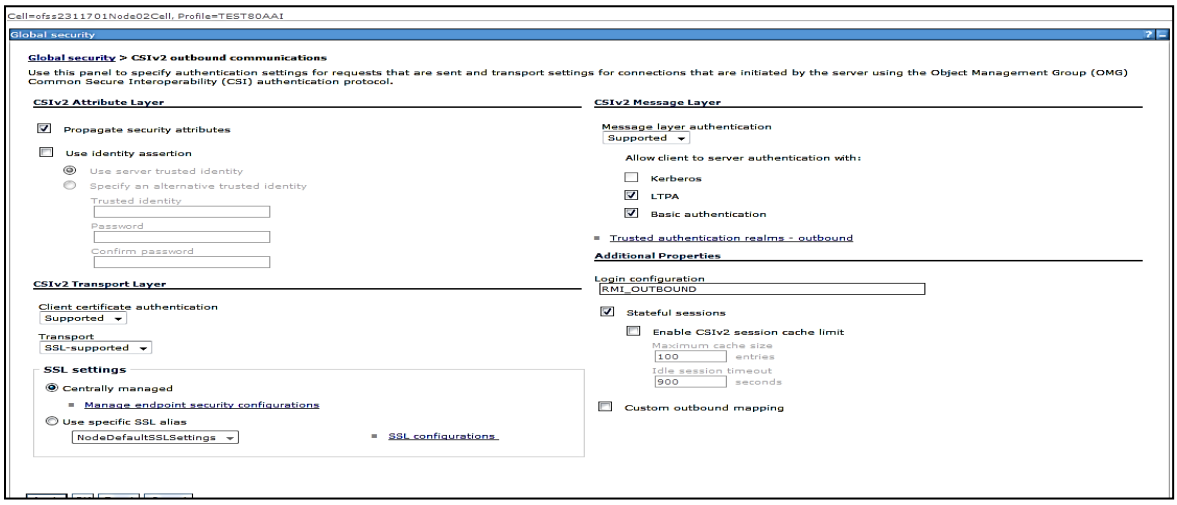


Figure 62:CSIv2 outbound communications

6. Click **Apply** and save details.

> **NOTE** RMI/IIOP Authentication Settings are not required for WebLogic.

4.2.9 Restart WebSphere Profile

For more information, refer to the Start/Stop Infrastructure Services section in the Oracle Financial Services Analytical Applications Infrastructure Installation and Configuration Guide available on the OHC page.

4.3 Configuring IPE in Tomcat

This section provides details for configuring IPE in Tomcat and includes the following topics:

- Prerequisites
- Configuring Tomcat

4.3.1 Prerequisites

The following are the prerequisites for Tomcat configuration:

- Before deployment, make required changes in App layer.

4.3.2 Configuring Tomcat

Perform the following procedure to configure Tomcat:

1. Create the datasource for ILP context in Tomcat by editing the `server.xml` file in the `<TOMCAT_HOME_DIR>/conf` directory.

> **NOTE** The following example is a sample configuration. Update your database details accordingly. Make sure the context name matches the directory name under webapps. In the following example, ILP is the context name

```
<Context path="/ILP" docBase="/scratch/ofsaobie/apache-tomcat-
8.0.21/webapps/ILP" debug="0" reloadable="false" crossContext="true">
        <Loader delegate="true"/>
    <Resource auth="Container"
                name="jdbc/FICMASTER"
                type="javax.sql.DataSource"
                driverClassName="oracle.jdbc.driver.OracleDriver"
                username="act_obiconf"
                password="password"
                url="jdbc:oracle:thin:@whf00aqr:1521/DEVUT08SPRINT"
```

```
                    maxTotal="100"
                    maxIdle="30"
                    maxWaitMillis="10000" removeAbandoned="true"
removeAbandonedTimeout="60" logAbandoned="true"/>
      <Resource auth="Container"
                    name="jdbc/OFSAAAIINFO"
                    type="javax.sql.DataSource"
                    driverClassName="oracle.jdbc.driver.OracleDriver"
                    username="act_obiatm"
                    password="password"
                    url="jdbc:oracle:thin:@whf00aqr:1521/DEVUT08SPRINT"
                    maxTotal="100"
                    maxIdle="30"
                    maxWaitMillis="10000" removeAbandoned="true"
removeAbandonedTimeout="60" logAbandoned="true"/>
     <Resource auth="Container"
                    name="jdbc/OFSAAAIINFOCNF"
                    type="javax.sql.DataSource"
                    driverClassName="oracle.jdbc.driver.OracleDriver"
                    username="act_obiatm"
                    password="password"
                    url="jdbc:oracle:thin:@whf00aqr:1521/DEVUT08SPRINT"
                    maxTotal="100"
                    maxIdle="30"
                    maxWaitMillis="10000" removeAbandoned="true"
removeAbandonedTimeout="60" logAbandoned="true"/>


      </Context>
```

2. Update the following parameters in the `install.properties` file in the `$FIC_HOME/ILP/conf` directory:

   `sql.config.datasource.jndi.name=java:comp/env/jdbc/FICMASTER`

   `sql.atomic.datasource.jndi.name=java:comp/env/jdbc/<RTI_INFODOM>`

   `sql.metadom.datasource.jndi.name=java:comp/env/jdbc/<RTI_INFODOM>CNF`

> **NOTE** Name should match the Resource Name defined in the `server.xml` file.

3. Update the `application-env.properties` file in `$FIC_HOME/ILP/conf` directory after commenting out the parameter

   ```
   spring.profiles.active=JMS,JMSApplicationCache,JMSGateway,JMSFeedBackGateway
   ```

   and adding

   ```
   spring.profiles.active=
   ```

   > **NOTE** The following is an example and the changes shown will remove the JMS dependencies for IPE.

   ```
   #
   # The spring profiles to activate. The list of available profiles are:
   # 1. JMS - To activate the use of JMS infrastructure. This is required for
   #          activating any other JMS profile
   # 2. JMSGateway - To activate JMS interface for receiving source entities
   # 3. JMSApplicationCache - To activate JMS for receiving cache refresh messages
   # 4. JMSFeedBackGateway - To activate the Feedback message from APPS like FCCM in JMS
   #
   #spring.profiles.active=JMS,JMSApplicationCache,JMSGateway,JMSFeedBackGateway
   spring.profiles.active=
   ```

4. Copy the following jars from the `FIC_HOME/realtime_processing/additionaljars/tomcat` directory to the `fichome/realtime_processing/WEB-INF/lib` directory:

   ```
   jms-api-1.1-rev-1.jar
   javax.ws.rs-api-2.0.1.jar
   ```

5. Execute the `ant.sh` script to create `ilp.ear` and `ilp.war` files from the `fichome/realtime_processing` directory.
6. Copy the `ilp.war` file to the `TOMCAT_HOME/webapps` directory (not required to explode).
7. Restart the Tomcat server.

4.4 Configuring IPE in Kafka

This section provides details for configuring IPE in Kafka and includes the following topics:

4.4.1 Prerequisites

The following are the prerequisites for Kafka configuration:

- Before deployment, make required changes in App layer.

4.4.2 Configuring Kafka

Perform the following procedure to configure Kafka:

1. Create the datasource for ILP context in Tomcat by editing the `server.xml` file in the `<TOMCAT_HOME_DIR>/conf` directory.

> **NOTE** The following example is a sample configuration. Update your database details accordingly. Make sure the context name matches the directory name under webapps. In the following example, ILP is the context name.

```
<Context path="/ILP" docBase="/scratch/ofsaobie/apache-tomcat-
8.0.21/webapps/ILP" debug="0" reloadable="false" crossContext="true">
         <Loader delegate="true"/>
     <Resource auth="Container"
                  name="jdbc/FICMASTER"
                  type="javax.sql.DataSource"
                  driverClassName="oracle.jdbc.driver.OracleDriver"
                  username="act_obiconf"
                  password="password"
                  url="jdbc:oracle:thin:@whf00aqr:1521/DEVUT08SPRINT"
                  maxTotal="100"
                  maxIdle="30"
                  maxWaitMillis="10000" removeAbandoned="true"
removeAbandonedTimeout="60" logAbandoned="true"/>
      <Resource auth="Container"
                  name="jdbc/OFSAAAIINFO"
                  type="javax.sql.DataSource"
                  driverClassName="oracle.jdbc.driver.OracleDriver"
                  username="act_obiatm"
                  password="password"
                  url="jdbc:oracle:thin:@whf00aqr:1521/DEVUT08SPRINT"
                  maxTotal="100"
                  maxIdle="30"
                  maxWaitMillis="10000" removeAbandoned="true"
removeAbandonedTimeout="60" logAbandoned="true"/>
      <Resource auth="Container"
                  name="jdbc/OFSAAAIINFOCNF"
                  type="javax.sql.DataSource"
                  driverClassName="oracle.jdbc.driver.OracleDriver"
                  username="act_obiatm"
                  password="password"
                  url="jdbc:oracle:thin:@whf00aqr:1521/DEVUT08SPRINT"
                  maxTotal="100"
                  maxIdle="30"
```

```
                    maxWaitMillis="10000" removeAbandoned="true"
removeAbandonedTimeout="60" logAbandoned="true"/>

      </Context>
```

2. Update the following parameters in the `install.properties` file in the `fichome/realtime_processing/conf` directory:

   ```
   sql.config.datasource.jndi.name=java:comp/env/jdbc/FICMASTER
   sql.atomic.datasource.jndi.name=java:comp/env/jdbc/<RTI_INFODOM>
   sql.metadom.datasource.jndi.name=java:comp/env/jdbc/<RTI_INFODOM>CNF
   ```

   NOTE Name should match the Resource Name defined in the `server.xml` file.

3. Update the `$TOMCAT_HOME/webapps/ILP/conf/install.properties` file for kafka broker details, the inbound topic names, and outbound topic names as shown in the following example:

   ```
   kafka.bootstrap-servers=whf00act.in.oracle.com:9092
   kafka.inbound.topic.group.id=IPE
   kafka.inbound.topic=IPERT.t
   kafka.outbound.topic=IPERESPONSE.t
   kafka.consumer.num=5
   kafka.inbound.partitions=10
   kafka.inbound.poll.timeout=3000
   ```

4. Update the `application-env.properties` file in `$TOMCAT_HOME/webapps/ILP/conf` directory after commenting out the parameter
   `spring.profiles.active=JMS,JMSApplicationCache,JMSGateway,JMSFeedBackGateway`
   and adding
   `spring.profiles.active=IPEKAFKA`

   NOTE The preceding example changes will remove the JMS dependencies for IPE and enable Kafka.

5. Update the `web.xml` file in the `$TOMCAT_HOME/webapps/ILP/WEB-INF` directory (append the `,/conf/applicationContext-kafka.xml` file in the end under **param contextConfigLocation**).

   ```
   /conf/applicationContext.xml,/conf/applicationContext-
   jms.xml,/conf/applicationContext-
   jmx.xml,/conf/ext/spring*.xml,/conf/applicationContext-kafka.xml
   ```

6. Copy the following jars from the `fichome/realtime_processing/additionaljars` directory to the `fichome/realtime_processing/WEB-INF/lib` directory:

a. spring-messaging-4.2.3.RELEASE.jar
b. spring-kafka-1.2.0.RELEASE.jar
c. spring-integration-kafka-2.0.1.RELEASE.jar
d. kafka-clients-0.11.0.0.jar
e. slf4j-log4j12-1.4.3.jar
f. slf4j-api-1.4.3.jar
g. jms-api-1.1-rev-1.jar
h. javax.ws.rs-api-2.0.1.jar

7. Update the following in the server.properties file under config directory:

```
num.partitions=200
auto.create.topics.enable=true
```

8. Execute the `ant.sh` script to create `ilp.ear` and `ilp.war` files from the `fichome/realtime_processing` directory.
9. Copy the `ilp.war` file to the `TOMCAT_HOME/webapps` directory.
10. Restart the Tomcat server to deploy `ilp.war`.

4.4.3 Testing Kafka Configuration

Perform the following procedure to test the Kafka configuration:

> **NOTE** The examples shown in this section display port number, URL, and request for illustrative purposes only. You must enter the correct values for the port number, URL and request as it exists in your application.

1. Use the following command to stop all Tomcat servers:

```
ss -tanp | grep 9092
Output: LISTEN     0      50                          :::9092
:::*      users:(("java",31114,554))
ps -eaf | grep  31114
kill -9 31114
```

> **NOTE** 9092 is the port number of the URL.

2. Use the following command to stop all Kafka servers:

```
ps -eaf | grep kafka
```

> **NOTE** CTRL+C will kill all kafka related process.

3. Open four putty sessions, one each for zookeeper, server, producer and consumer. Enter the following commands in the respective sessions after changing the path to kafka:
   - Zookeeper session: `bin/zookeeper-server-start.sh config/zookeeper.properties`
   - Server session: `bin/kafka-server-start.sh config/server.properties`
   - Producer session: `bin/kafka-console-producer.sh --broker-list whf00bfk.in.oracle.com:9092 --topic IPERT.t`
   - Consumer session: `bin/kafka-console-consumer.sh --bootstrap-server whf00bfk.in.oracle.com:9092 --topic IPERESPONSE.t`
4. Start Tomcat.
5. Enter the following request in the producer window in a single line:

```
"eg:    { ""type"" : ""DIM_ACCOUNT"",""domain"" : ""FRA"",""appId"" : ""OFS_IPE"",""runtype"" : 1,""runParam"" : 1,""attributes"" : {""Account Skey"" : ""500"",""Account Description"" : ""ASD"" },""additionalParams"" : { }}
"
```

6. Check for the results in the Consumer window.
7. Check for results in **rti_eval_assmnt_result** and **rti_assmnt_result**.

5 Post data in JSON format to IPE

JSON requests are posted through REST services in IPE Realtime Mode. The subsections in this topic provides information on how to post data in JSON format using HTTP requests and receive responses from the server.

5.1 Prerequisites

The following are the prerequisites to post JSON requests in IPE:

1. Install a JSON client on your local computer.
2. User access for authentication on the server.
3. Knowledge about posting JSON requests.

5.2 Posting JSON request for REST services

The following is the procedure to post JSON requests for REST services:

1. Open the JSON client.
2. Select or enter Method as **POST**.
3. Enter the server URL in **Request URL**.
4. Enter the Header parameters as shown in the following table:

Table 11: JSON Request Header Parameters

No.	Header Name	Header Value	Description
1	username	Enter the user name to login to the server.	This value is used for user authentication.
2	password	Enter the password to login to the server.	This value is used for user authentication.
3	content-type	Select or enter application/json.	This value denotes that the data in JSON format.
4	accept	Select or enter application/json.	This value denotes that the data in JSON format

5. Select **application/json** for **Body content type**.
6. Enter the JSON message in the body. The following code sample is an example:

```
{
  "type": "DIM_ACCOUNT",
  "domain": "FRA",
```

```
  "appId": "OFS_IPE",
  "runtype": 1,
  "runParam": 1,
  "attributes": {
    "Account Skey": 181,
    "Account Description": "ABC"
  },
  "additionalParams": {}
}
```

The following is the convention for the data in the fields of the preceding code sample:

```
{
  "type": <ACTIVITY Table>,
  "domain": <Processing Segment>,
  "appId": <Product ID>,
  "runtype": 1,
  "runParam": 1,
  "attributes":
{
    <Required attributes/Business Column names of activity table>:
<Values>
  },
  "additionalParams": {}
}
```

7. Send the JSON message from the client to the server. The server provides a response message with values for successful posting or for errors, if any.

6 Additional Configuration

To use IPE framework without a Sample Application, perform the following additional configuration.

6.1 Updating Install Properties

To update the install properties, follow these steps:

1. Login to the UNIX machine where the OFS AAAI Application Pack is installed with IPE enabled.
2. Navigate to <OFSAA Installation Directory>/realtime_processing/WebContent/conf.
3. Edit `install.properties` and provide the following information:

Table 12: Additional configuration for without sample application

Parameter Code	Value	Description
<RTI_INFODOM>	< Infodom Name >	Infodom Name used for IPE For example, OFSAAAIINFO
<RTI_SEGMENT>	< default IPE processing segment code>	Enter the default IPE Processing Segment Code. For example, FRA

6.1.1 Updating install.properties to enable Highlights in Real-Time

The following is an example of the `install.properties` file with Highlights in Real-Time disabled. To enable, set the parameter `ipe.produce.hglights.results=false` to `ipe.produce.hglights.results=true` in the file.

```
sql.config.datasource.jndi.name=jdbc/FICMASTER
sql.atomic.datasource.jndi.name=jdbc/OFSAAAIINFO
sql.metadom.datasource.jndi.name=jdbc/OFSAAAIINFOCNF
system.infodom=OFSAAAIINFO
system.domain=FRA
system.appid=OFS_IPE
deployment.assessment.execution.mode=LIVE
deployment.datastore=RDBMS
deployment.test.java.naming.initial.context.factory=
deployment.test.java.naming.provider.url=
batch.audit.backup=true
ipe.persist.eval.results=false
process.maxalert.count=100
action.json.constant=200
action.json.response.url=
```

```
aai.auth.url=
kafka.bootstrap-servers=
kafka.inbound.topic.group.id=IPE
kafka.inbound.topic=IPERT.t
kafka.outbound.topic=IPERESPONSE.t
ipe.produce.hglights.results=false
kafka.consumer.num=5
kafka.inbound.partitions=10
kafka.inbound.poll.timeout=3000
```

6.1.2 IPE Caching

To perform caching in IPE, follow these steps:

1. Connect to OFSAA Configuration schema.
2. Replace the placeholders with the following information in the update queries.
3. Run the update queries.

Parameter Code	Value	Description
<INITIALFACTORY>	< initial context for app server>	Initial Context for the Web Application Server Websphere: com.ibm.websphere.naming.WsnInitialContextFactory Weblogic: weblogic.jndi.WLInitialContextFactory
<PROVIDER_URL>	<processing URL>	URL for accessing Queues and Topics in Web Application server. For more information, refer to Appendix A.

4. Update configuration c set `c.PARAMVALUE='<INITIALFACTORY>'` where `c.paramname ='RTI_PROCESSING_INITIALCONTEXTFACTORY'`
   `/`
   Update configuration c set `c.PARAMVALUE='<PROVIDER_URL>'` where `c.paramname ='RTI_PROCESSING_PROVIDER_URL'`
   `/`

6.2 Creating ILP.ear/ ILP.war

It is mandatory to have the ILP.ear in the same profile or domain where the `<contextname>.ear` file of the OFS AAAI Application is deployed. To create ILP.ear/ ILP.war, follow these steps:

1. Navigate to < OFSAA Installation Directory >/realtime_processing.
2. Execute the command:

   ```
   ./ant.sh.
   ```

3. On successful execution, the ILP.ear and ILP.war files are generated under the <OFSAA Installation Directory >/realtime_processing folder.

6.3 Deploying ILP.ear in Weblogic

This section defines how to deploy ILP.ear in Weblogic.

NOTE

1. It is mandatory to have ILP.ear in the same domain where <contextname>.ear of the OFS AAAI Application is deployed.
2. o not deploy **ilp.ear** on Sanctions TFLT, the installation will fail.

To deploy ILP.ear in Weblogic, follow these steps:

1. Start the Weblogic server.
2. Create an ILP.ear folder in <WEBLOGIC_INSTALL_DIR>/user_projects/domains/<DOMAIN_NAME>/applications
3. Copy <`FIC_HOME>/realtime_processing/ILP.ear` to <WEBLOGIC_INSTALL_DIR>/user_projects/domains/<DOMAIN_NAME>/applications/ILP.ear/.
4. Explode the ILP.ear file by executing the command:

   ```
   jar -xvf ILP.ear
   ```

5. Delete the `ILP.ear` and IPL.war files.
6. Create an `ILP.war` folder in <WEBLOGIC_INSTALL_DIR>/user_projects/domains/<DOMAIN_NAME>/applications`/ILP.ear`
7. Copy <`FIC_HOME>/realtime_processing/ILP.war` to <WEBLOGIC_INSTALL_DIR>/user_projects/domains/<DOMAIN_NAME>/applications/ILP.ear/ILP.war
8. Explode the `ILP.war` file by executing the command:

   ```
   jar –xvf ILP.war.
   ```

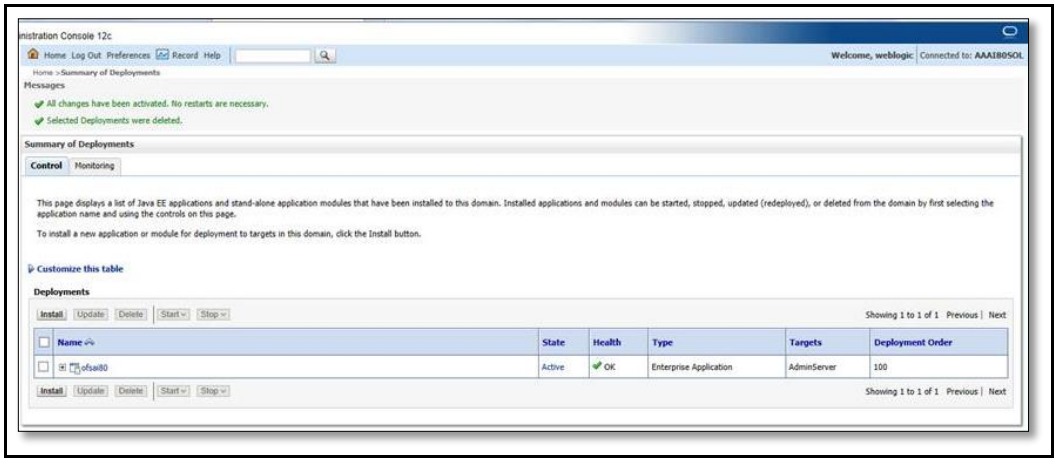
9. Delete the ILP.war file.

6.3.1 Installing ILP.ear in WebLogic using WebLogic Administrator Console.

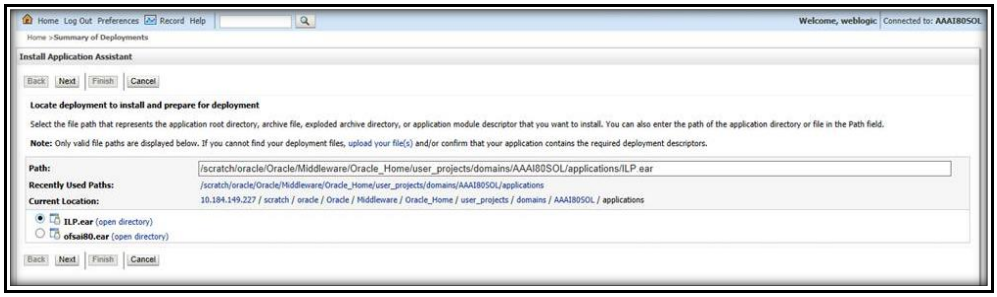
1. Navigate to the path <WEBLOGIC_INSTALL_DIR>/user_projects/domains/<DOMAIN_NAME>/`bin` in the machine in which WebLogic is installed.
2. Start WebLogic by executing the command:

   `./startWebLogic.sh -d64 file`

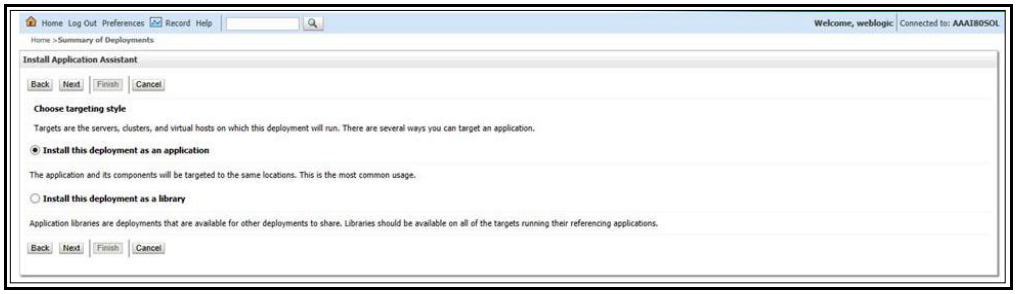
3. Open the following URL in the browser window: `http://<ipaddress>:<administrative console port>/console`. (Use https protocol if SSL is enabled). The *Welcome* window is displayed.
4. Login with the Administrator **Username** and **Password**. The Summary of Deployment page is displayed.



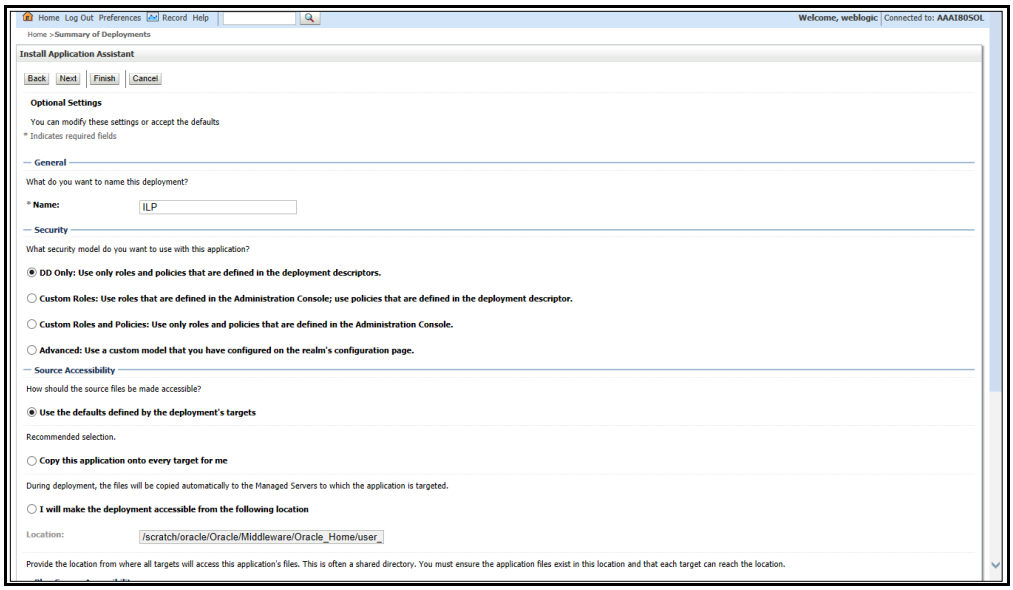
5. Click **Install**. The Install Application Assistance page is displayed.



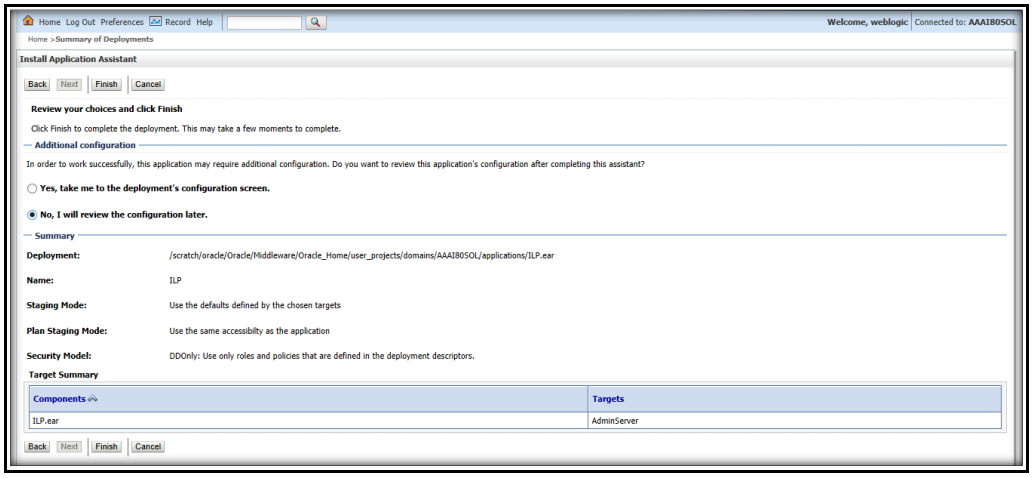
6. Select **ILP .ear** and click **Next**. The Install Application Assistance page is displayed with the Choose targeting style section.



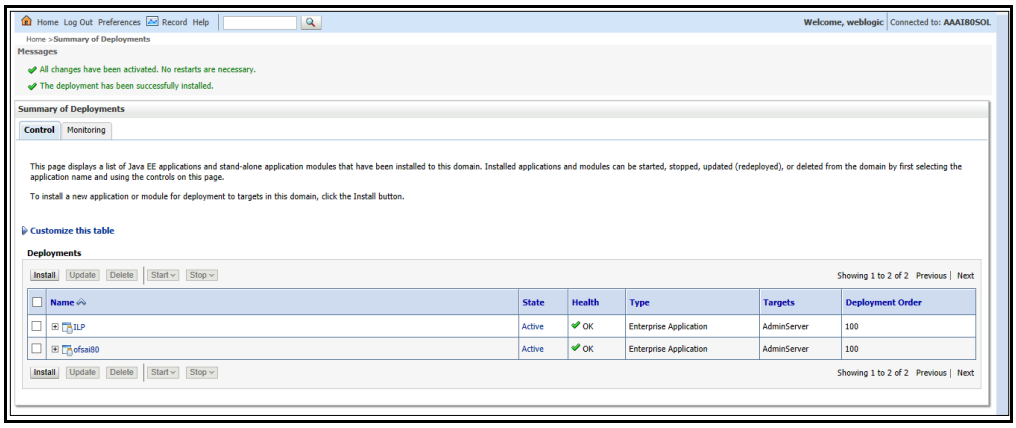
7. By default, the **Install this deployment as an application** option in the Choose targeting style section is selected. Click **Next**. The Install Application Assistance page is displayed with the Optional Settings section.



8. Retain the default selections and click **Next**. The Install Application Assistance page is displayed with the Review your choices and click Finish section.



9. Select **No, I will review the configuration later** in the Additional Configuration section and click **Finish**. ILP is added in the Name section of the Summary of Deployment page with following message: *The deployment has been successfully installed.*



10. Restart all OFS AAAI servers. For more information, refer to the Start/Stop Infrastructure Services section in the Oracle Financial Services Analytical Applications Infrastructure Installation and Configuration Guide available on the OHC page.

6.3.2 Deploying ILP.ear in WebSphere

To deploy ILP.ear. in WebSphere, follow these steps:

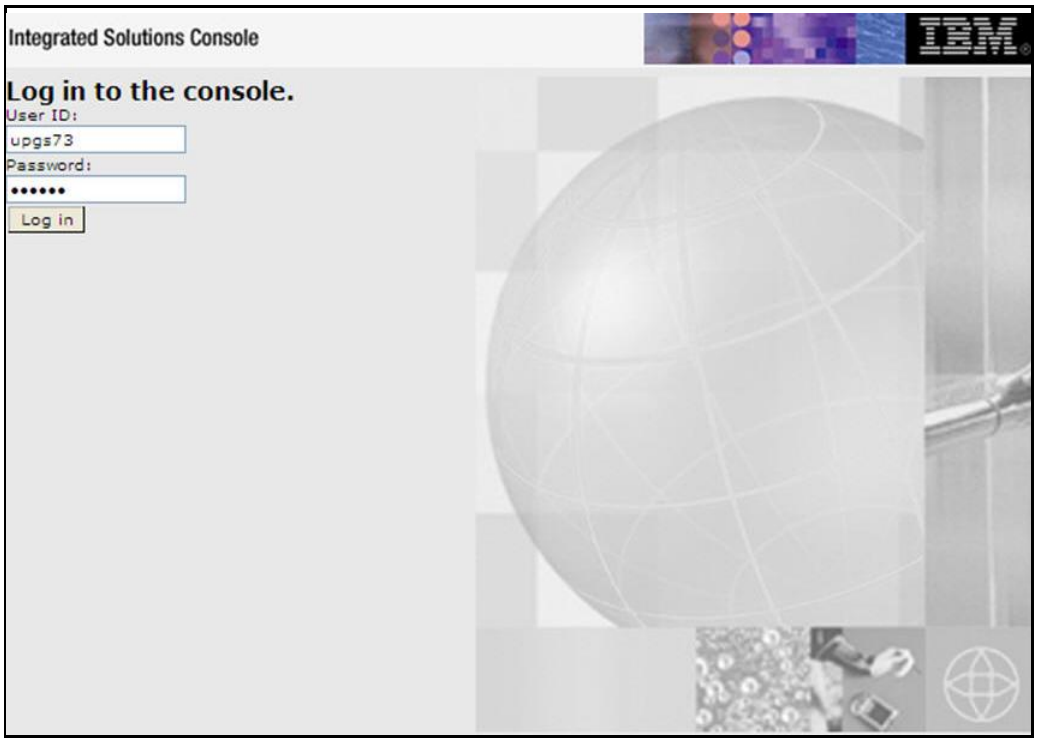
NOTE

1. It is mandatory to have ILP.ear in the same profile where <contextname>.ear of OFS AAAI Application is deployed.
2. Do not deploy **ilp.ear** on Sanctions TFLT, the installation will fail

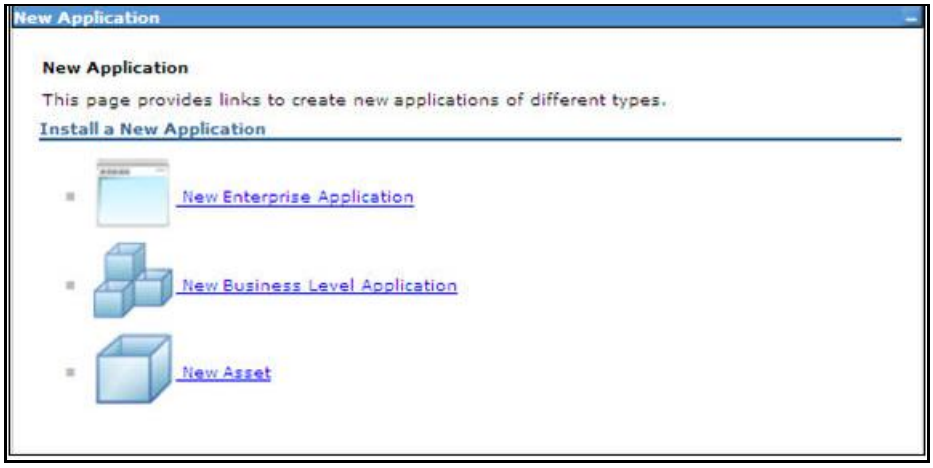
1. Start the WebSphere Profile by navigating to the path "/<Websphere_Installation_Directory>/IBM/WebSphere/AppServer/profiles/<Profile_Name>/bin/" then execute the command:

   ./startServer.sh server1

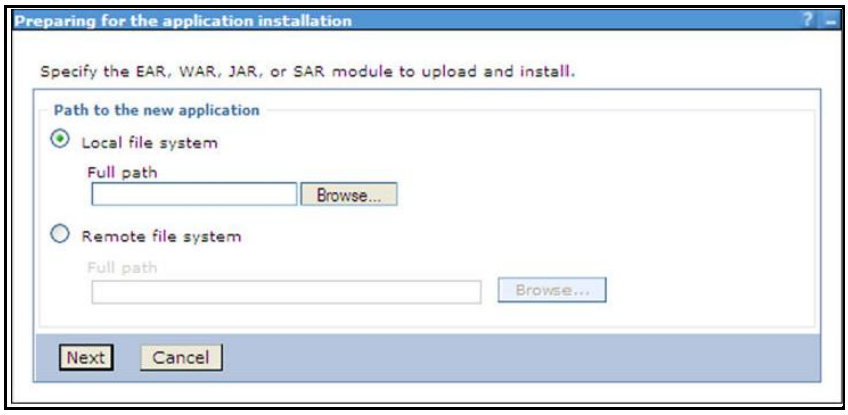
2. Open the following URL in the browser: `http://<ipaddress>:<Administrative Console Port>/ibm/console`. (use https protocol if SSL is enabled). The login screen is displayed.



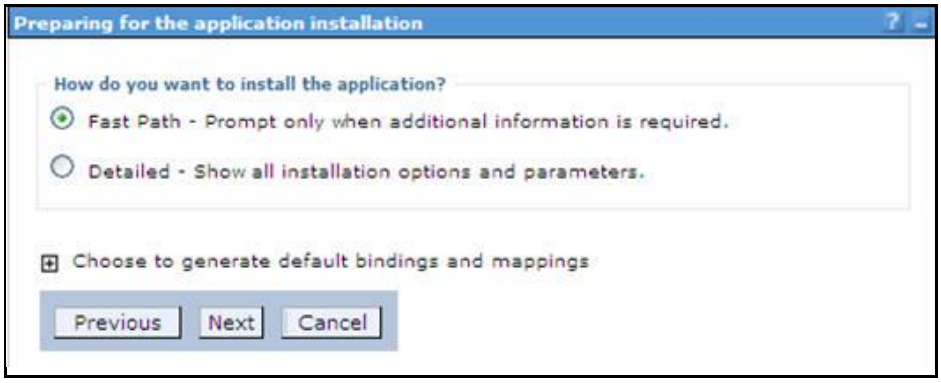
3. Enter the user credentials which have administrator rights and click **Log In**.
4. From the LHS menu, select **Applications** and click **New Application**. The *New Application* window is displayed.



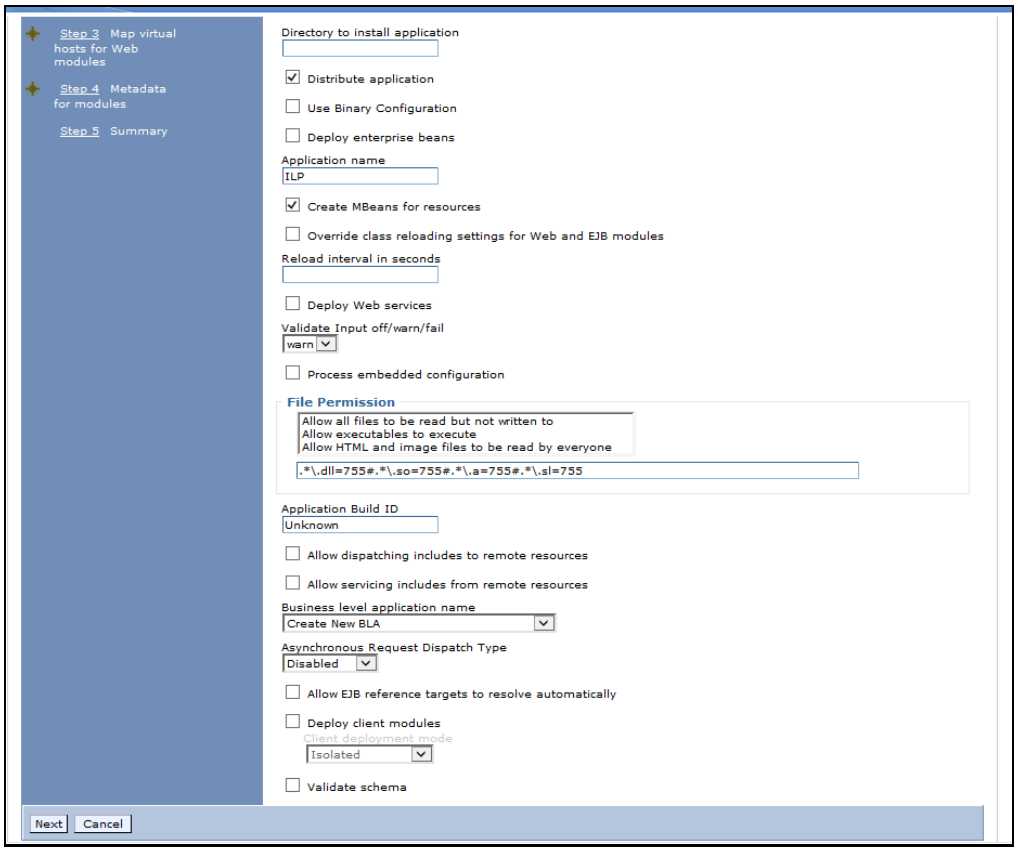
5. Click **New Enterprise Application**. The *Preparing for the application installation* page is displayed.



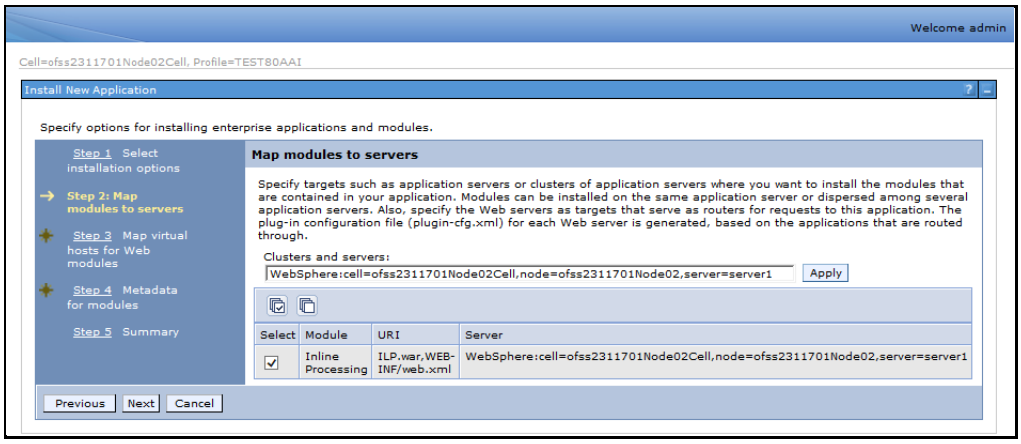
6. Select **Remote File System** and click **Browse**. Select the EAR file generated for IPE to upload and install. Click **Next**.



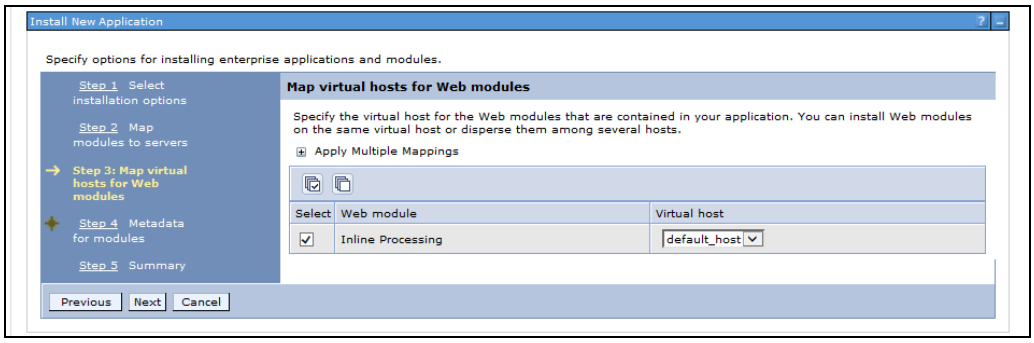
7. Select the **Fast Path** option and click **Next**. The *Install New Application* window is displayed.



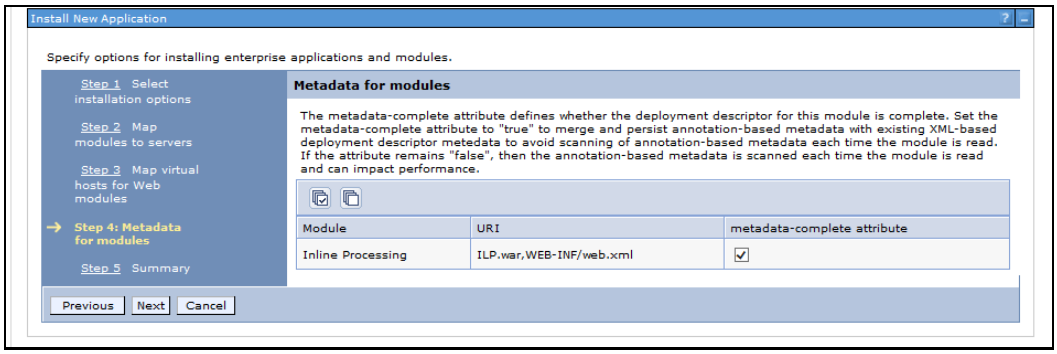
8. Click **Next**. The *Map Modules to Servers* page is displayed.



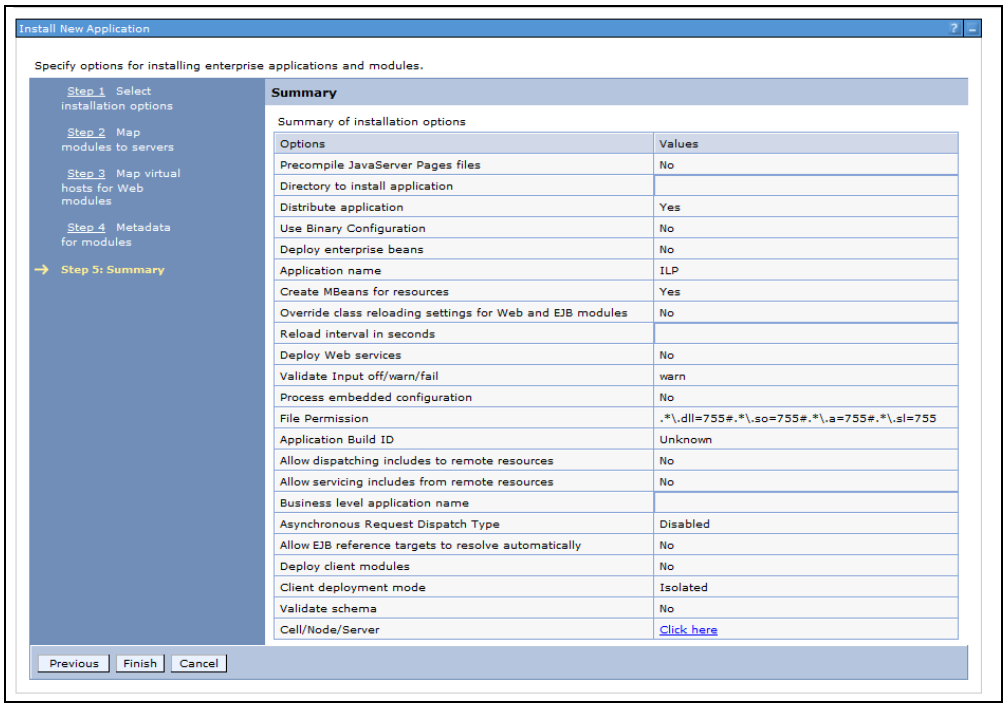
9. Select the **Inline Processing** checkbox and click **Next**. The Map Virtual hosts for Web modules page is displayed.



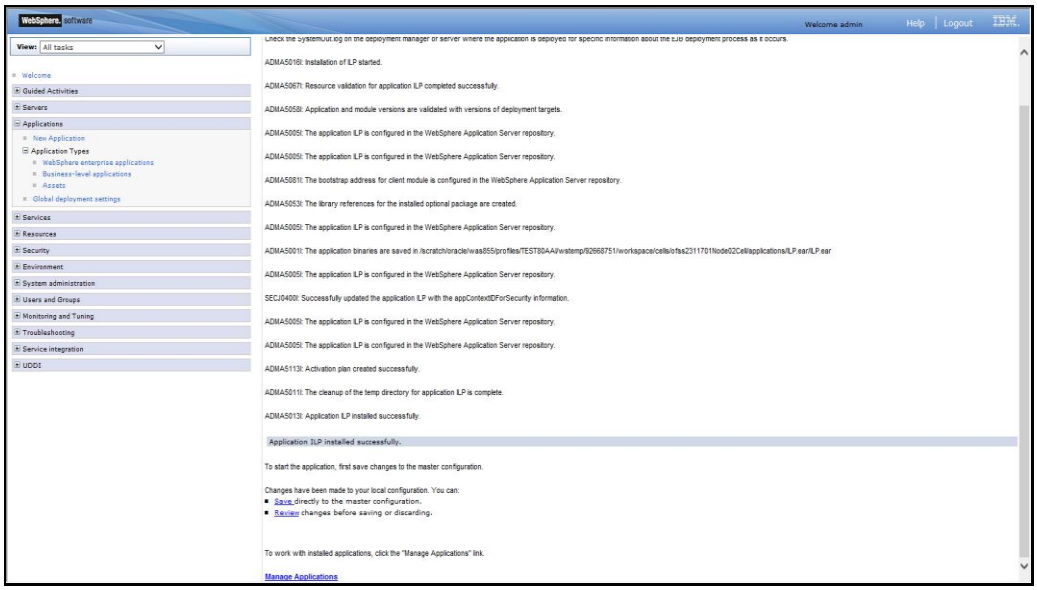
10. Select the **Inline Processing** checkbox and click **Next**. The Metadata for modules page is displayed.



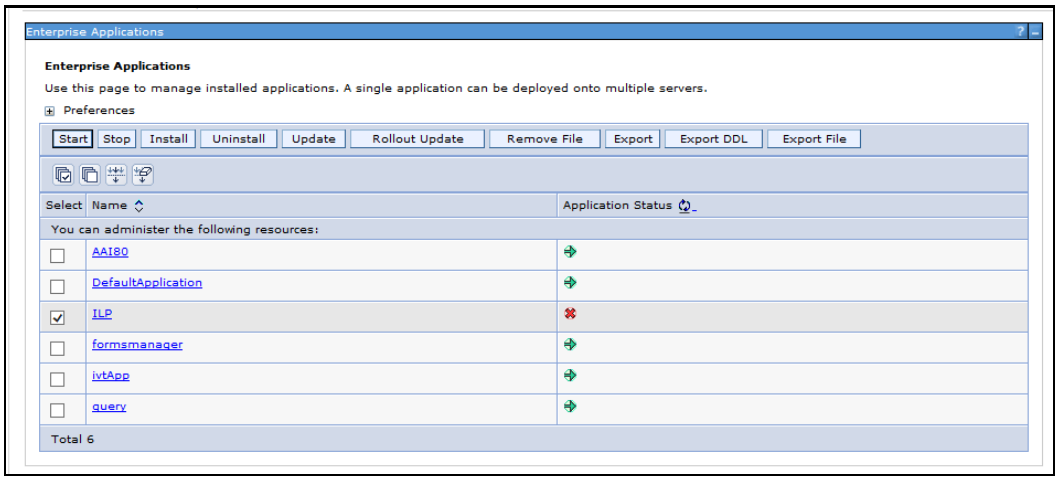
11. Select the **Metadata-complete** attribute checkbox and click **Next**. The Summary page is displayed.



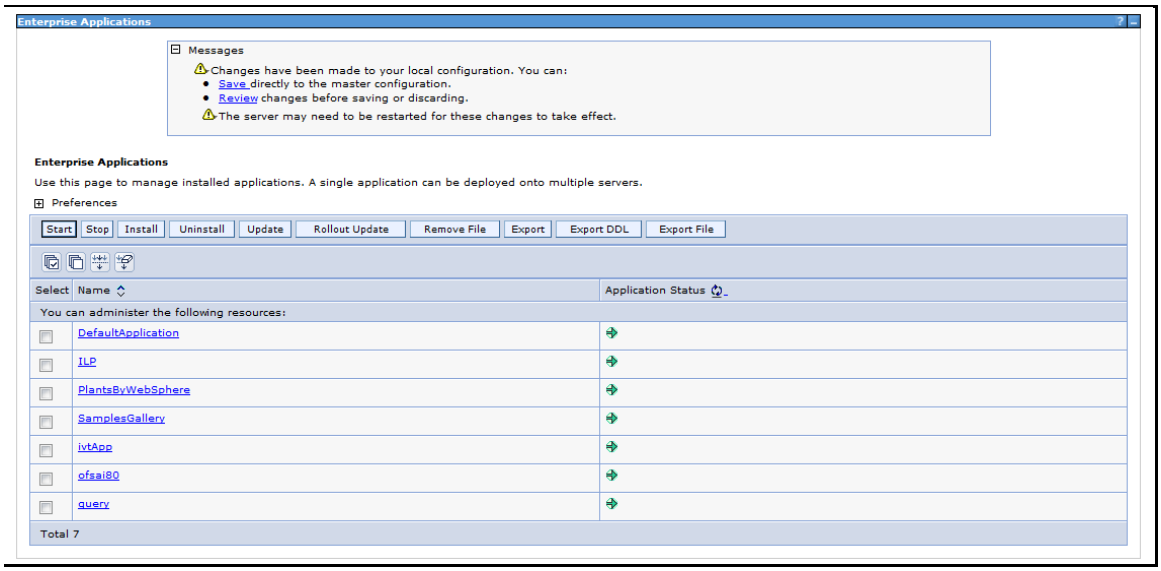
12. Click **Finish**. On successful installation, a message is displayed.



13. Click **Save** and save the master file configuration. The details are displayed in the *Master File Configuration* page.



14. Select **ILP** and click **Start**. The Enterprise Application page is displayed with confirmation message.



15. Restart all OFS AAAI servers. For more information, refer to the Start/Stop Infrastructure Services section in the Oracle Financial Services Analytical Applications Infrastructure Installation and Configuration Guide available on the OHC page.

6.4 Logging

Once the sample application client is triggered, the logs get written onto RTI-server.log from the path `<<ILP deploy area>>/logs`

By default, the log mode will be set to 'info'. If detailed logs are required then follow these steps:

1. Navigate to <FIC_HOME>/realtime_processing/WebContent/WEB-INF/
2. Edit the file `log4j.xml` to modify the level value of the code from 'info' to 'debug' as follows:

```
<logger name="com.ofs.aai">
<level value="DEBUG"/>
</logger>
```

3. Create `ilp.ear/ilp.war`. For more information, refer to the *Creating ILP.ear/ ILP.war* section in the *Oracle Financial Services Inline Processing Engine Configuration Guide* available on the OHC page.
4. Deploy the `ILP.ear` file. For more information, refer to the *Deploying ILP.ear. in Weblogic section* or *Deploying ILP.ear in Websphere* section in the Oracle Financial Services Inline Processing Engine Configuration Guide available on the OHC page.

7 HIVE Configurations

To run IPE in HIVE, perform the following configurations:

> **NOTE** HIVE supports only batch mode processing.

7.1 Loading UDF file in HIVE

To load the UDF file in HIVE, follow these steps:

1. Copy the JAR file `$FIC_HOME/realtime_processing/ipeudf /lib/ofsaa_ipe_udf.jar` to the host on which HIVEServer2 is running.
2. Provide the HIVE user with Read, Write and Execute permissions to this directory, and make a note of the path (For example, `/opt/local/hive/lib/`).
3. Login to the Cloudera Manager Console as an admin user and navigate to Clusters menu, click on HIVE. The HIVE service window appears.
4. Click the **Configuration** tab.

> **NOTE** This is the Classic layout of the Cloudera Manager console.

5. Click Service-Wide menu and select **Advanced**.
6. Configure the HIVE Auxiliary JARs Directory property with the HIVEServer2 host path from Step 1, `/opt/local/hive/lib/`.
7. Click **Save Changes**. The JARs are added to `HIVE_AUX_JARS_PATH` environment variable.
8. Redeploy the HIVE client configuration. Follow these steps:
   i. Navigate to the HIVE service in the Cloudera Manager Admin Console.
   j. On the Actions menu, select Deploy Client Configuration.
   k. Click Deploy Client Configuration.
9. Restart the HIVE service. If the HIVE Auxiliary JARs Directory property is configured but the directory does not exist, HIVEServer2 does not start. Follow these steps:
10. On the **Actions** menu, select **Restart.**
11. Click **Restart**.

7.2 Creating Result Tables

To create Result Tables for HIVE, follow these steps:

1. Login to the server where OFSAA is installed.
2. Navigate to the path `$FIC_HOME/realtime_processing/infodomscripts/DDL/hive` and execute the scripts in `create_infodom.hql` file into the Datadom (HIVE schema).
3. The tables `RTI_ASSMNT_EVAL_RESULT` and `RTI_ASSMNT_RESULT` are created.

8 Appendix A

8.1 Inline Processing URL of JMS

Construct URL for JMS as below.

- **Websphere→**iiop://<APP_SERVER_HOST_NAME>:<BOOTSTRAP_ADDRESS>

  `For example, iiop://myhost.mydomain.com:2809`

> **NOTE** For more information, refer to the *Oracle Financial Services Analytical Applications Infrastructure Inline Processing Configuration Guide* available in the OTN page. Check the port in the *Websphere* section.

- **Weblogic→**t3://<APP_SERVER_HOST_NAME>:< <SERVLET PORT> (Use t3s protocol for SSL)

  `For example, t3://myhost.mydomain.com:7001`

> **NOTE** `<APP_SERVER_HOST_NAME>` is the IP or the Host Name of a server where WebLogic or WebSphere is installed.

9 Appendix B

9.1 Check ports in WebSphere

To check the ports in WebSphere, follow these steps:

1. Open the following URL in the browser window: http://<ipaddress>:<administrative console port>/ibm/console. (https if SSL is enabled). The Login window is displayed.
2. Login with the Administrator **Username** and **Password**.
3. Click **+** to expand **Servers**.
4. Click **+** to **Server Types**.
5. Click **WebSphere application servers**.

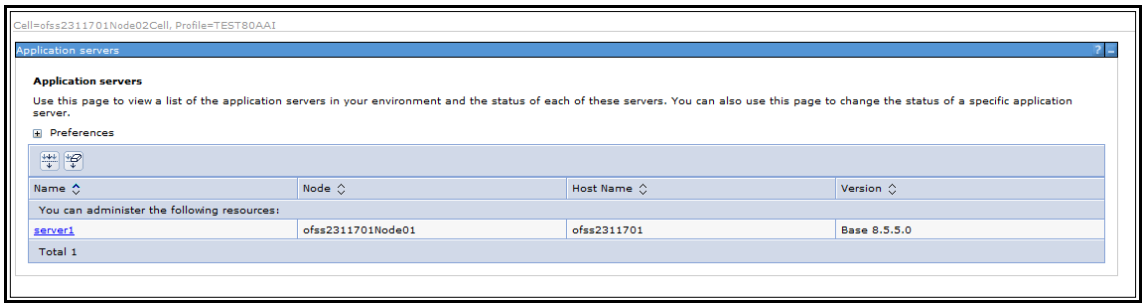


Figure 63: Application Servers

6. Click **server1**.

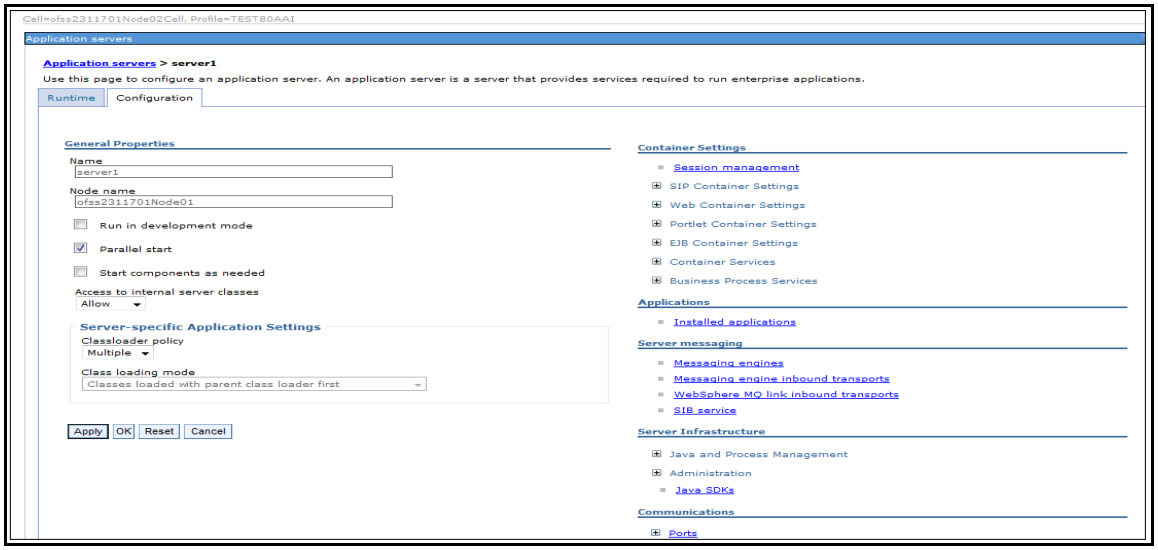


Figure 64: Application Servers

7. Click + to expand **Ports** under **Communications**.

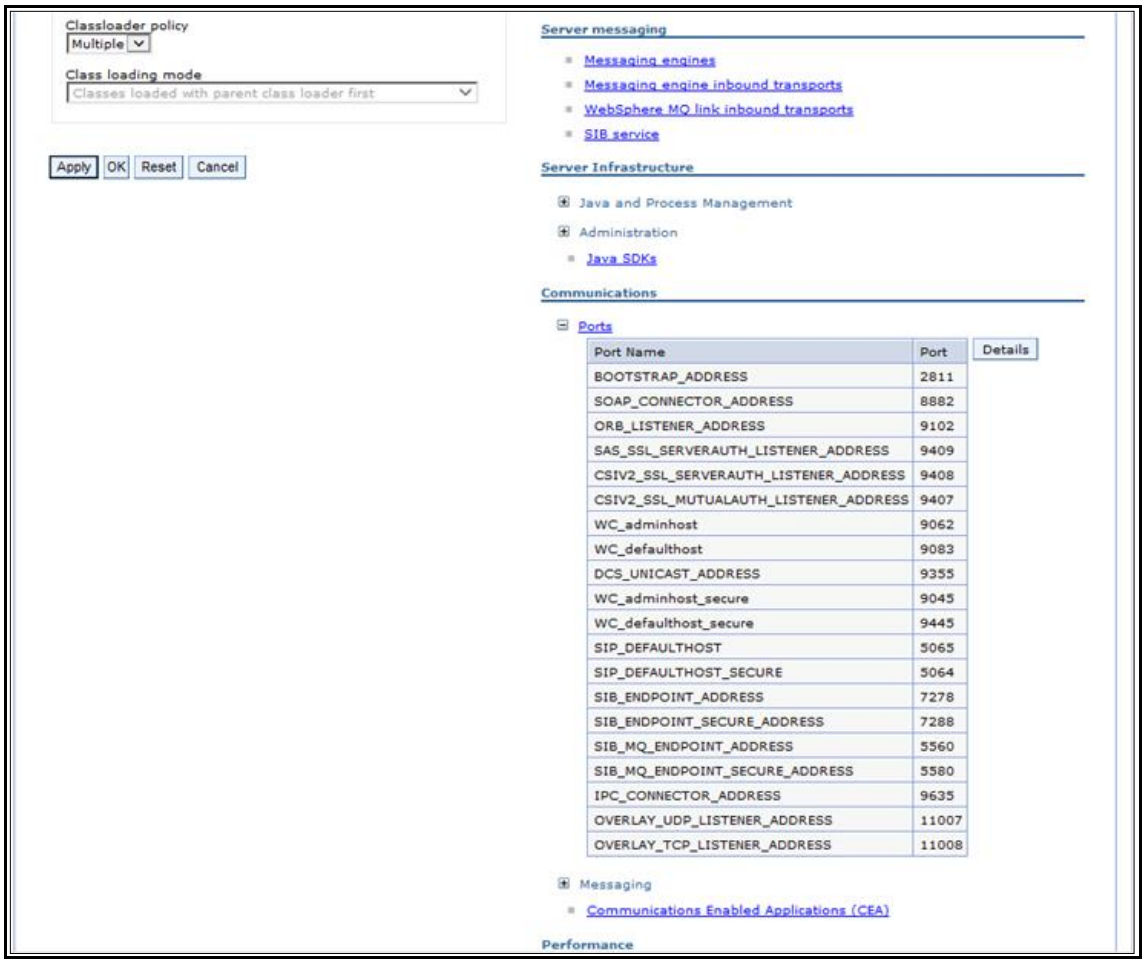


Figure 65: Ports List

10 Appendix C

10.1 Checking Target Inbound transport chain and Provider endpoints values

To check the values, follow these steps:

1. Open the following URL in the browser window: http://<ipaddress>:<administrative console port>/ibm/console. (https if SSL is enabled). The Login window is displayed.
2. Login with the Administrator **Username** and **Password**.
3. Click **+** to expand **Servers** in the LHS menu.
4. Click + Server Types.
5. Click **WebSphere application servers**. The Application servers screen is displayed.

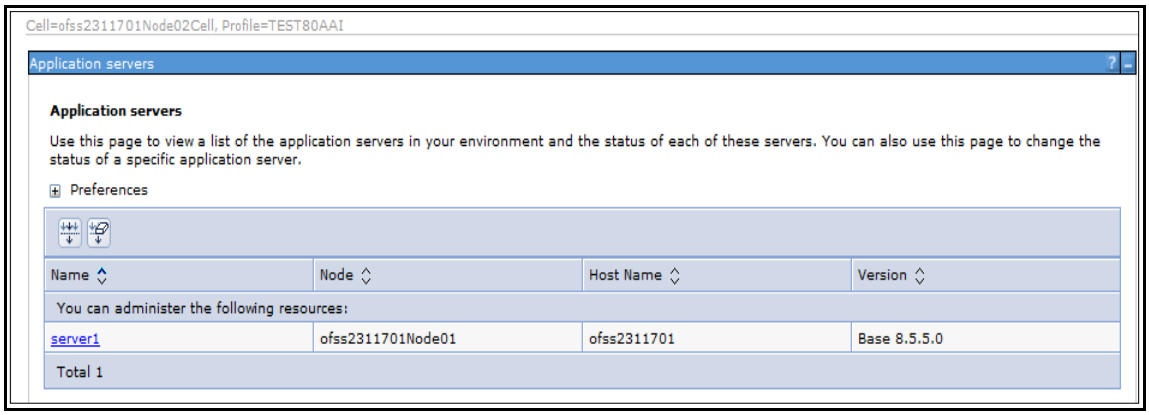


Figure 66: Application servers

6. Click **server1**.
7. Navigate to **Configuration** tab.

Figure 67: Configuration

8. Under **Server messaging**, select **Messaging engine inbound transports**. The *Transport Chain* screen is displayed.

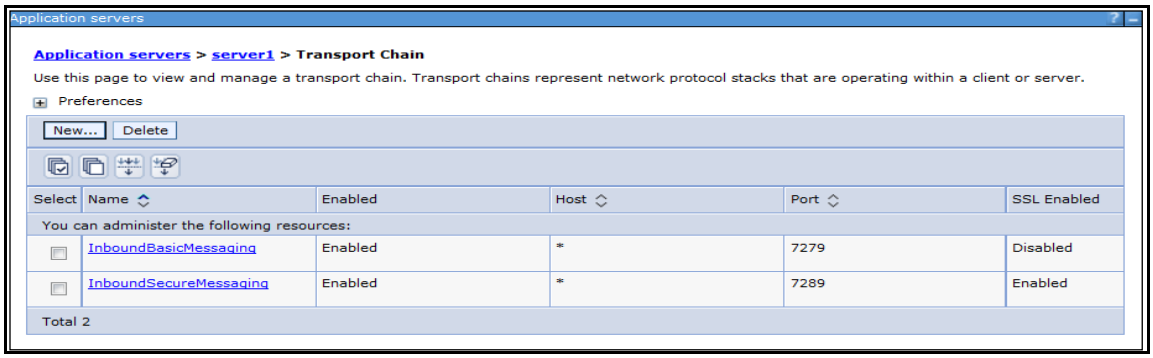


Figure 68: Transport Chain

9. Note the Transport chain name InboundBasicMessaging for Target Inbound Transport Chain.
10. Use the following **Provider endpoints** format: <WebSphere_HostName> : <SIB_ENDPOINT_ADDRESS port>: <Transport Chain Name>
    - <WebSphere_HostName>: The hostname of the server where WebSphere is installed.
    - <SIB_ENDPOINT_ADDRESS port>: The transport chain port corresponding for Transport chain name as InboundBasicMessaging.
    - <Transport Chain Name>: The Transport chain name as InboundBasicMessaging.

    For example: ofssxxxxx.in.oracle.com:7279:InboundBasicMessaging

> **NOTE** The transport chain name and Provider endpoints should be entered during configuration of JMS Connection Factory. Refer to section Configuring JMS Connection Factory for more details.

Send Us Your Comments

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, indicate the title and part number of the documentation along with the chapter/section/page number (if available) and contact the Oracle Support.

Before sending us your comments, you might like to ensure that you have the latest version of the document wherein any of your concerns have already been addressed. You can access My Oracle Support site which has all the revised/recently released documents.