

Setting Up JMS (Java Message Service) Applications

At the end of this module, you will be able to:

- ✓ Understand how WebLogic Server JMS is implemented
- ✓ Configure JMS administered objects using the administration console
- ✓ Configure persistent messages
- ✓ Use the WLS administration console to monitor JMS

1. WebLogic Server JMS Administration

- Messaging Fundamentals
- Point-to-Point (PTP) and Publish-Subscribe (Pub/sub) domains
- Configuring JMS Objects
- Fine-Tuning WLS JMS

2. Configuring Persistent Messaging

3. Monitoring JMS in WLS

Message-Oriented Middleware

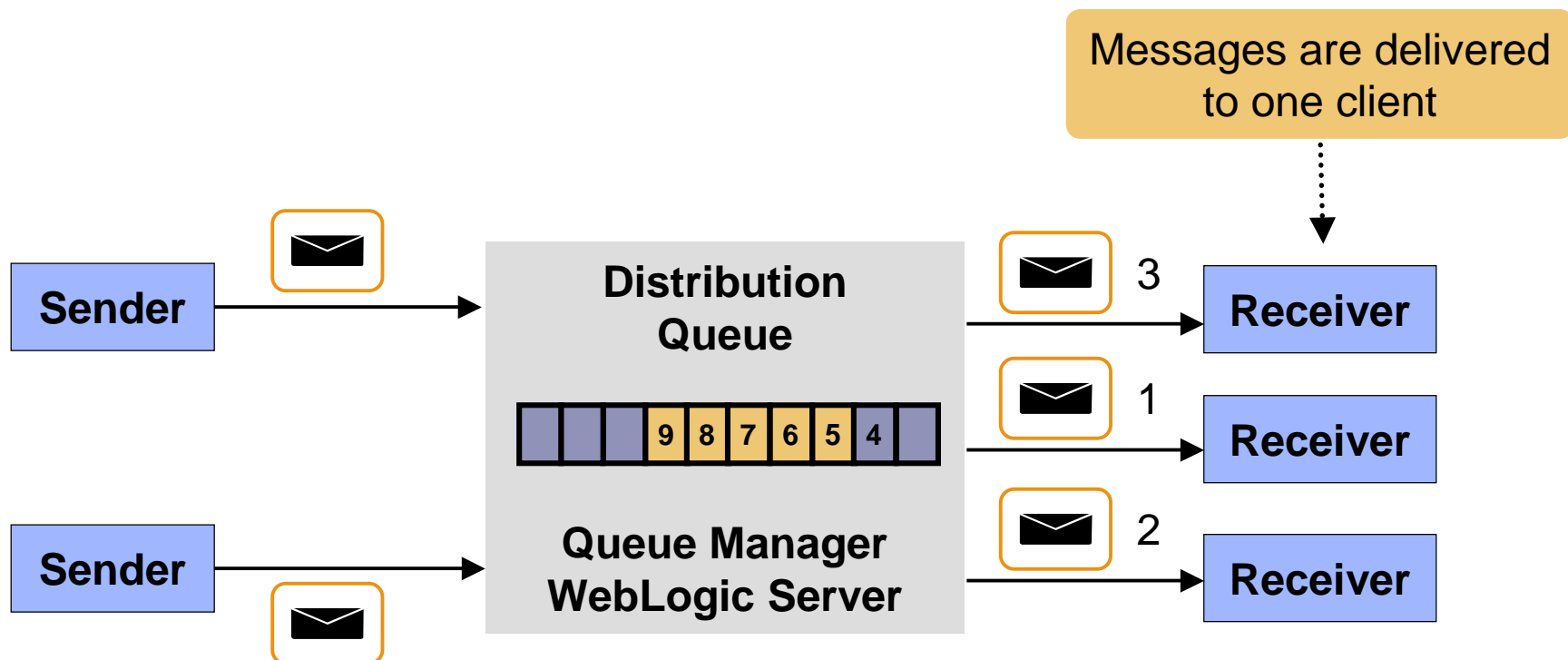


- ▶ *Message-oriented middleware* refers to an infrastructure that supports messaging.
- ▶ Typical message-oriented middleware architectures define these elements:
 - Message structure
 - The way to send and receive messages
 - Scaling guidelines

Point-to-Point (PTP) Queue

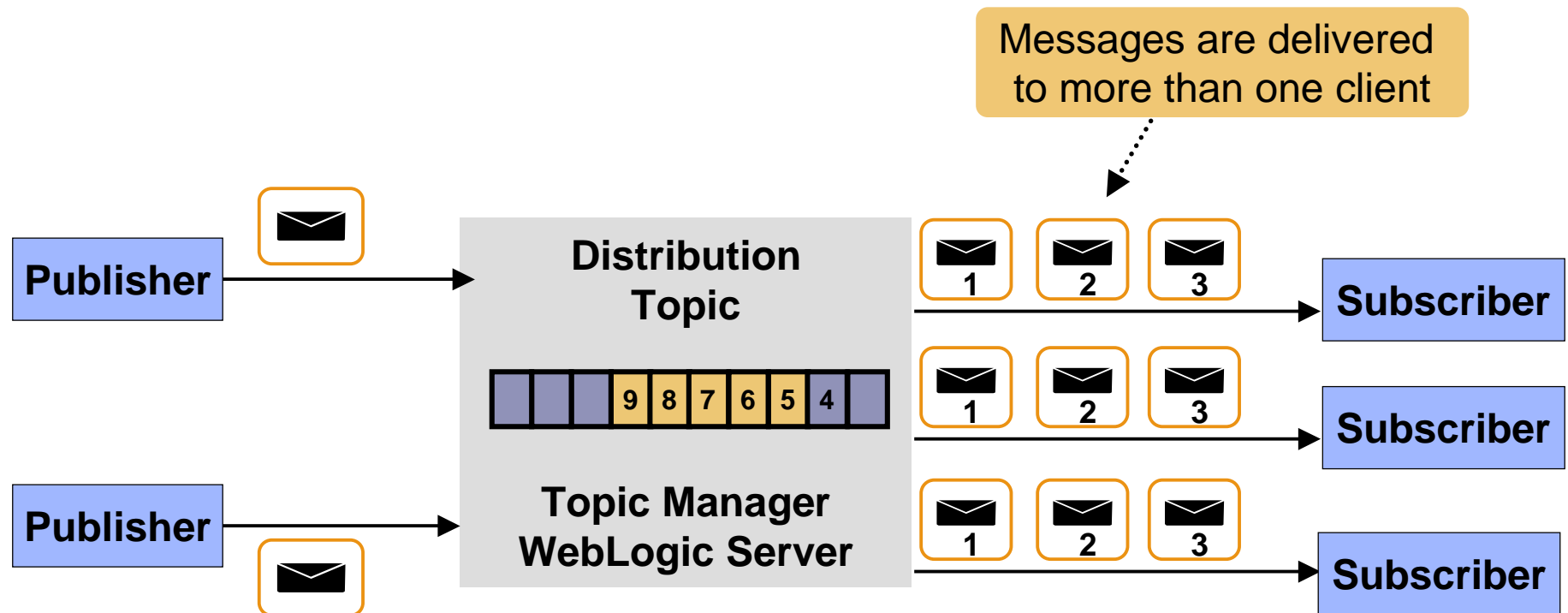


- Many producers can *serialize* messages to multiple receivers in a *queue*.



Publish-Subscribe Topics

- Publishing and subscribing to a *topic* decouples producers from consumers.

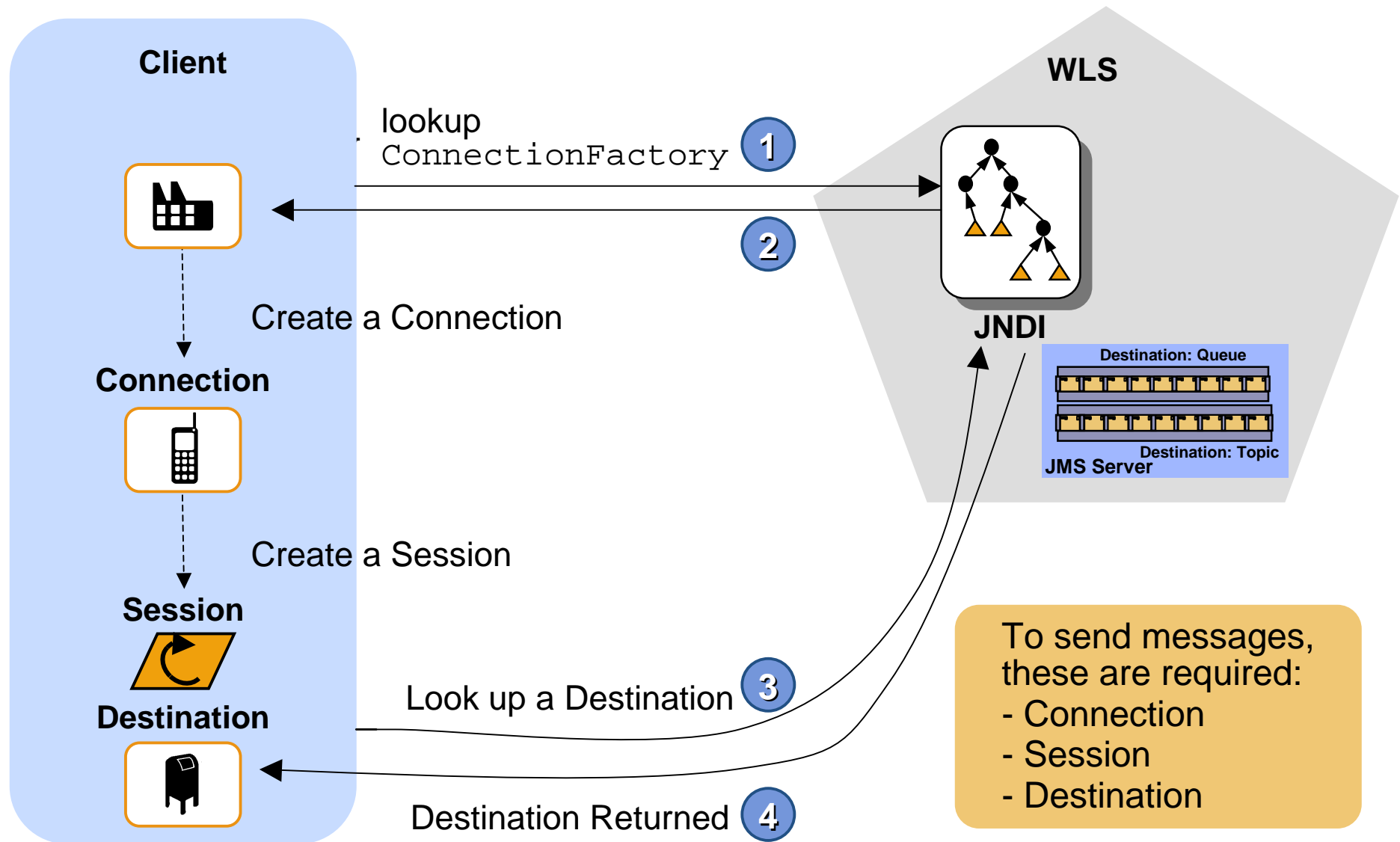


WebLogic Server JMS Features



- ▶ WebLogic Server JMS supports:
 - PTP and Pub/sub domains
 - Guaranteed and transactional message delivery
 - Durable subscribers
 - Distributed destinations
 - Recovery from failed servers

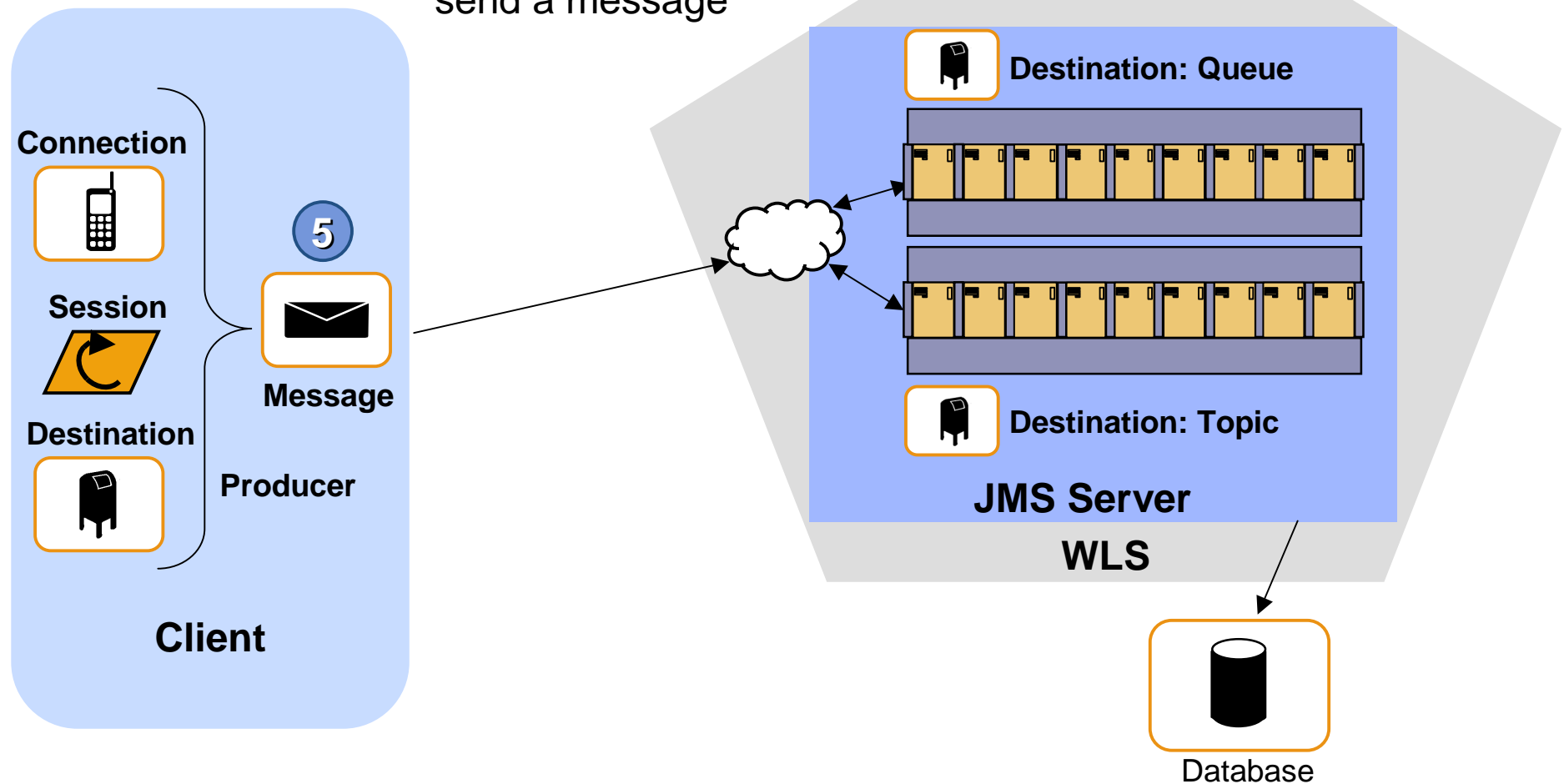
JMS Architecture: Connecting



JMS Architecture: Sending Messages



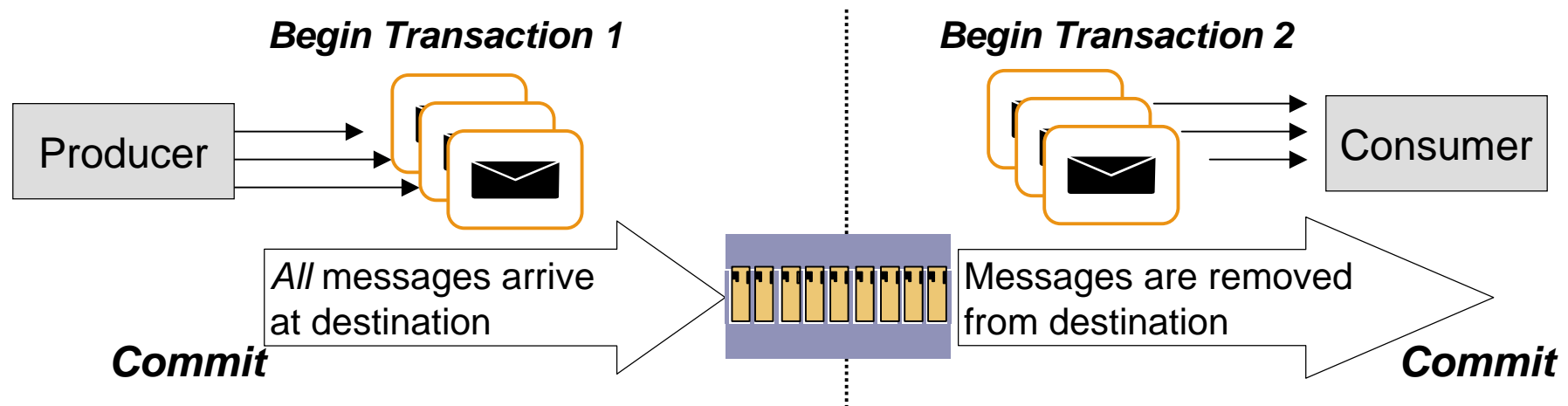
Connection, Session and Destination are used to send a message



Transacted Messaging



- ▶ A JMS client can use JTA to participate in a distributed transaction.
- ▶ Alternatively, a JMS client can demarcate transactions local to the JMS Session, through a transacted session.
- ▶ Participation in a transaction is optional.



Administrative Tasks

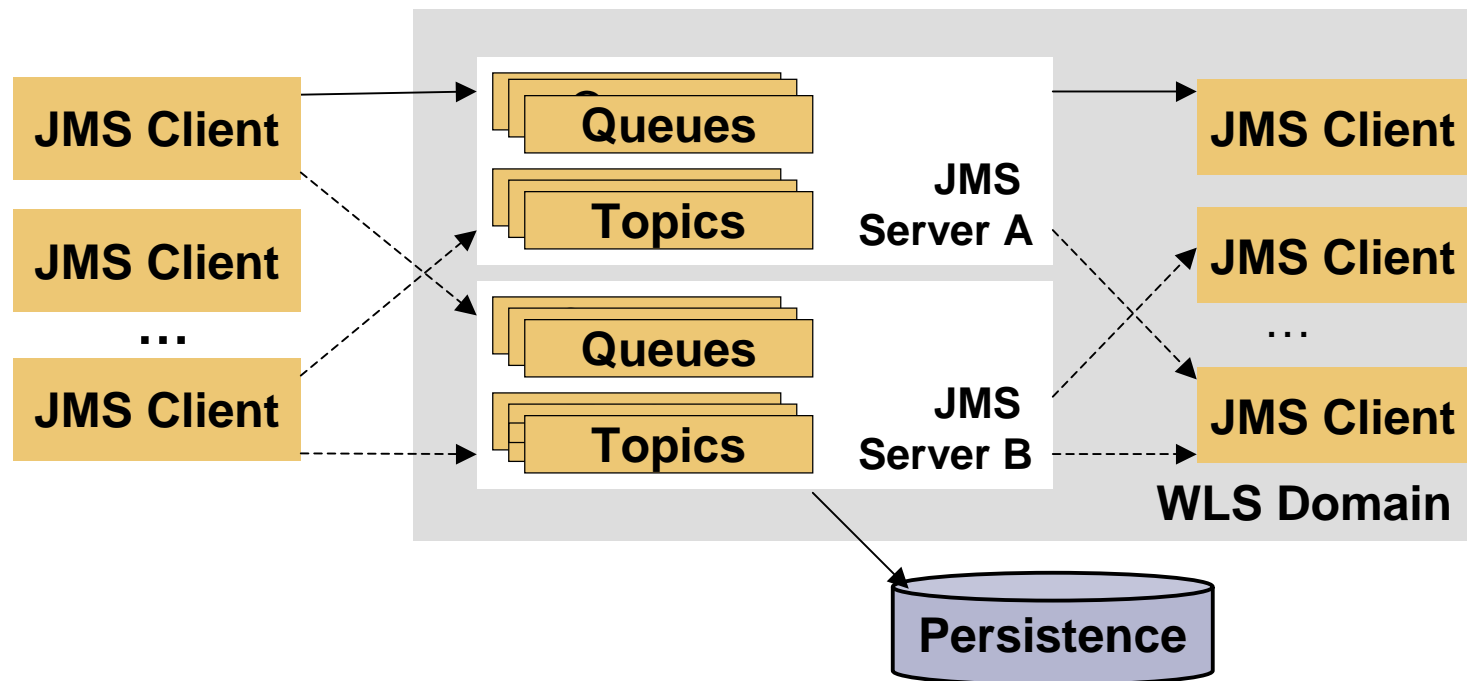


- ▶ Administrative tasks include these:
 - Creating and monitoring JMS Servers
 - Creating connection factories
 - Creating and monitoring destinations
 - Creating JMS stores
 - Configuring thresholds and quotas
 - Configuring durable subscriptions
 - Managing JMS service fail-over

WLS JMS Server



- ▶ In WLS, the messaging service is implemented through a JMS Server.
- ▶ A JMS Server receives and distributes messages.



Create a JMS Server



JMS Servers

New (2) Delete

Showing 1 - 2 of 2 Previous | Next

<input type="checkbox"/> Name	Persistent Store	TargetName
<input type="checkbox"/> examplesJMSServer	exampleJDBCStore	examplesServer
<input type="checkbox"/> WseeJMSServer	WseeFileStore	examplesServer

New Delete

Showing 1 - 2 of 2 Previous | Next

Create a New JMS Server

Back Next (4) Finish Cancel

JMS Server Properties

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

What would you like to name your new JMS Server?



Name:

MyJMS Server

Specify persistent store for the new JMS Server.

Persistent Store:

(none)

Create a New Store

Target a JMS Server



Create a New JMS Server

Back Next Finish Cancel

Select targets **2**

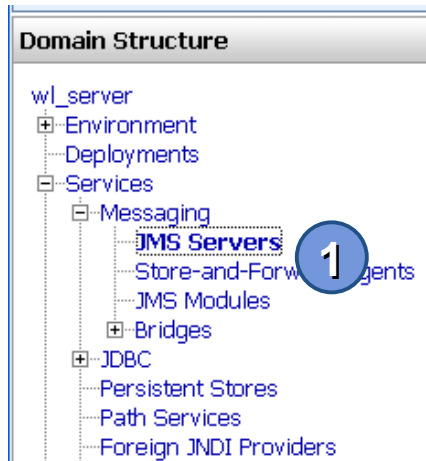
Select the server instance or migratable target on which you would like to deploy this JMS Server.

Target:

None
None
examplesServer **1**

Back Next Finish Cancel

Configure a JMS Server



Customize this table

JMS Servers

New Delete		Showing 1 - 2 of 2 Previous Next	
<input type="checkbox"/>	Name	Persistent Store	TargetName
<input type="checkbox"/>	examplesJMSServer	exampleJDBCStore	examplesServer
<input type="checkbox"/>	WseeJMSServer	WseeFileStore	examplesServer
New Delete		Showing 1 - 2 of 2 Previous Next	

Configuration Logging Targets Monitoring Control Notes

General Thresholds and Quotas Session Pools

Save

JMS servers act as management containers for the queues and topics in JMS modules that are targeted to them. A JMS server's primary responsibility for its destinations is to maintain information on what persistent store is used for any persistent messages that arrive on the destinations, and to maintain the states of durable subscribers created on the destinations.

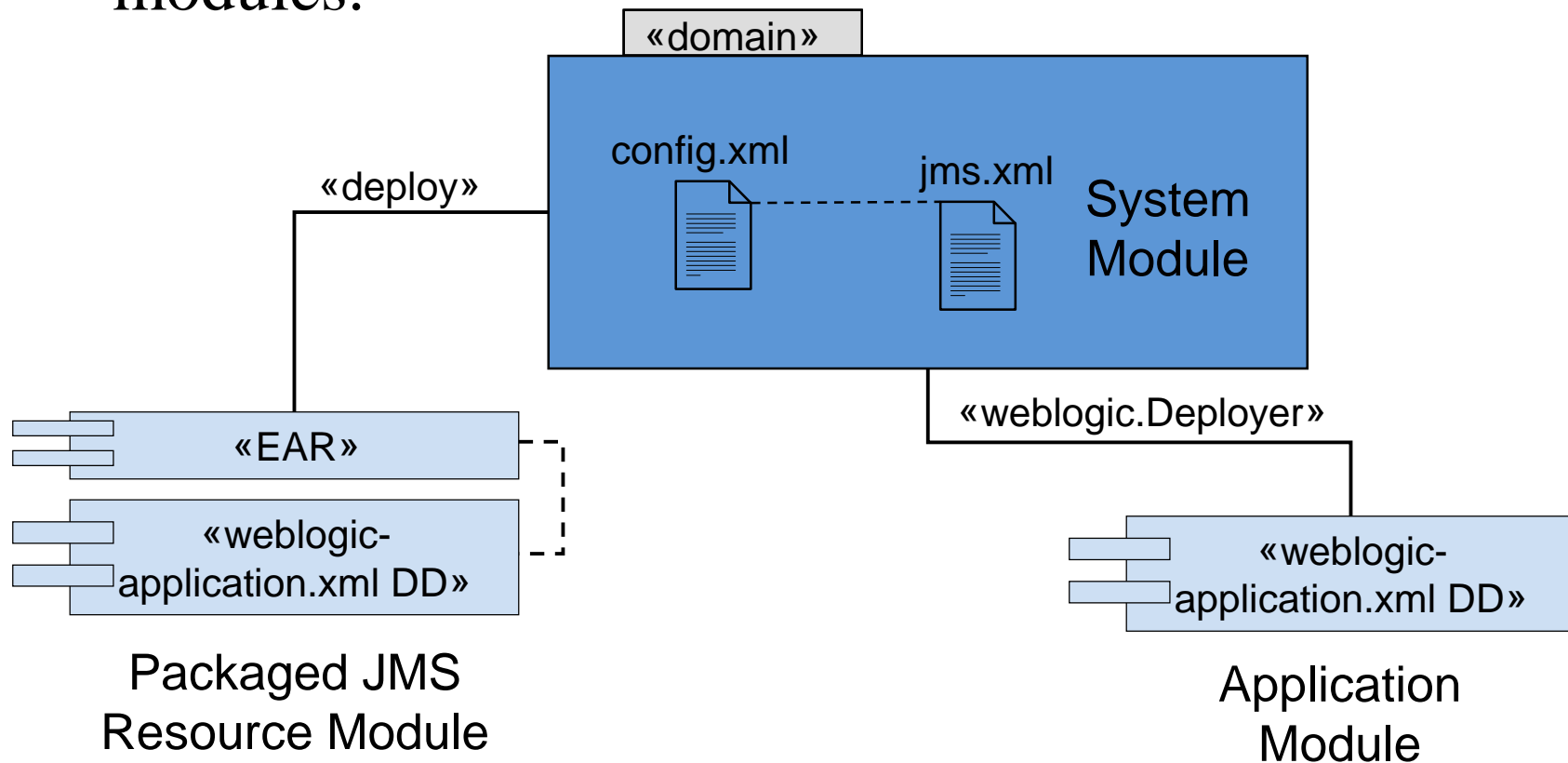
Use this page to define the general configuration parameters for this JMS server.

	Name:	examplesJMSServer	The name of this JMS server. More Info...
	Persistent Store:	<input type="text" value="exampleJDBCStore"/>	The file or database in which this JMS server stores persistent messages. If unspecified, the JMS server uses the default persistent store that is configured on each targeted WebLogic Server instance. More Info...
	Paging Directory:	<input type="text"/>	Specifies where message bodies are written when the size of

JMS Resources



- JMS resources are managed as system modules, application modules, or packaged JDBC resource modules.



Modular JMS Resource Configuration and Deployment...



- ▶ JMS configurations in WebLogic Server are stored as modules
 - Defined by an XML file that conforms to the `weblogic-jmsmd.xsd` schema
 - Similar to standard J2EE modules
- ▶ An administrator can create and manage JMS modules as:
 - Global system resources
 - Global standalone modules
 - Modules packaged with an enterprise application

...Modular JMS Resource Configuration and Deployment



- ▶ An advantage of modular deployment is simplified migration between environments, such as:
 - From development to integration
 - From system test to production
- ▶ You can migrate your application *and* the required JMS configuration:
 - Without opening an EAR file
 - Without extensive manual JMS reconfiguration

Connection Factory



- ▶ *A connection factory:*
 - Encapsulates connection configuration information
 - Is used to create pre-configured connections
 - Is stored in JNDI
 - Can be targeted to servers or clusters
- ▶ WLS provides a default connection factory that is bound in JNDI to `weblogic.jms.ConnectionFactory`.
- ▶ When a new configuration is required, a new connection factory can be created.

Create a Connection Factory...



Customize this table

JMS Modules

New Delete Showing 1 - 1 of 1 Previous | Next

<input type="checkbox"/>	Name	Type
<input type="checkbox"/>	examples-jms	System

New Delete Showing 1 - 1 of 1 Previous | Next

Summary of Resources

3 New Delete Showing 1 - 8 of 8 Previous | Next

<input type="checkbox"/>	Name	Type	JNDI Name	Targets
<input type="checkbox"/>	exampleQueue	Queue		examplesJMSServer
<input type="checkbox"/>	exampleTopic	Topic		examplesJMSServer
<input type="checkbox"/>	exampleTopic	ConnectionFactory	weblogic.examples.jms.TopicConnectionFactory	examplesServer
<input type="checkbox"/>	exampleTrader	ConnectionFactory	jms.connection.traderFactory	examplesServer
<input type="checkbox"/>	jms/MULTIDATASOURCE_MDB_QUEUE	Queue		examplesJMSServer
<input type="checkbox"/>	quotes	Topic	quotes	examplesJMSServer
<input type="checkbox"/>	weblogic.examples.jms.QueueConnectionFactory	ConnectionFactory	weblogic.examples.jms.QueueConnectionFactory	examplesServer
<input type="checkbox"/>	weblogic.wsee.wseeExamplesDestinationQueue	Queue	weblogic.wsee.DefaultQueue	WseeJMSServer

New Delete Showing 1 - 8 of 8 Previous | Next

...Create a Connection Factory...



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 topics, foreign servers, and JMS SAF destinations, you can also proceed to targeting pages for selecting appropriate server target mechanism for grouping JMS module resources and the members to server resources.

4



Connection Factory

Defines a set of connection configuration parameters that are used to create



Queue

Defines a point-to-point destination type, which are used for asynchronous p



Topic

Defines a publish/subscribe destination type, which are used for asynchronou

...Create a Connection Factory...



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 dizzyworldJMSModule.

What would you like to name your new connection factory?

Name:

6

dizzyworldConnectionFa

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



JNDI Name:

7

dizzyworldConnectionFa

Back

Next

Finish

Cancel

8

...Create a Connection Factory...



Create a New JMS System Module Resource

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

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

Use this page to view and accept the default targets where this JMS resource will be targeted. The default targets are based on **Advanced Targeting** to use the subdeployment mechanism for targeting this resource.

The following JMS module targets will be used as the default targets for your new JMS system module resource. If the module'

Targets:

Servers

☒ dizzy1

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

9

...Create a Connection Factory



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 you can create a new subdeployment by clicking the **Create a New Subdeployment** button. You can also reconfigure subdeployment

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

Subdeployments:

11

(none)
(none)
dizzyworldSubdeployment
dizzyworldClusterSubdeployment

[Create a New Subdeployment](#)

What targets do you want to assign to this subdeployment?

Targets:

Configure Connection Factory: Default Delivery



Configuration	Subdeployment	Notes				
General	Default Delivery	Client	Transactions	Flow Control	Load Balance	Security

Save

Use this page to define the default delivery configuration parameters for this JMS connection factory, such as the default delivery

Default Priority:	<input type="text" value="4"/>	The default priority used for messages when a priority
Default Time-to-Live:	<input type="text" value="0"/>	The maximum length of time, in milliseconds, that a r 0 indicates that the message has an infinite amount tir
Default Time-to-Deliver:	<input type="text" value="0"/>	The delay time, in milliseconds, between when a mess
Default Delivery Mode:	<input type="text" value="Persistent"/>	The default delivery mode used for messages when a
Default Compression Threshold:	<input type="text" value="2147483647"/>	The number of bytes for the serialized message body received by the JMS message producer or consumer. f
Send Timeout:	<input type="text" value="10"/>	The maximum length of time, in milliseconds, that a se the message being sent. More Info...
Default Unit-of-Order for Producer:	<input type="text" value="None"/>	The default Unit-of-Order name for producers that cor order, even among multiple recipients. More Info...
User-generated Unit-of-Order Name:	<input type="text"/>	Specifies the Unit-of-Order name when the Default Un

Configure Connection Factory: Client



Configuration	Subdeployment	Notes				
General	Default Delivery	Client	Transactions	Flow Control	Load Balance	Security

Save

Use this page to define the client configuration parameters for this JMS connection factory, such as client id for durable subscribers.

Client ID for Durable Subscribers:	<input type="text"/>	An optional client ID for a durable subscriber. One JMS client from using a connection factory can have only one durable subscriber with the same client ID. More Info...
<input checked="" type="checkbox"/> Allow Close() Within onMessage()		Specifies whether the connection factory allows a client to call Close() within the onMessage() method. More Info...
Client Acknowledge Policy:	All ▼	Acknowledge policy for non-transacted messages. All acknowledges all unacknowledged messages up to, and including, the message being acknowledged. More Info...
Maximum Messages per Session:	10	The maximum number of messages that can be received by a synchronous consumer in Synchronous Prefetch Mode is enabled. More Info...
Prefetch Mode for Synchronous Consumer:	Disabled ▼	Specifies whether a synchronous consumer can prefetch messages. More Info...
Multicast Overrun Policy:	Keep Old ▼	The policy to use when the number of messages in the multicast buffer exceeds the maximum number of messages. More Info...

Configure Connection Factory: Transactions



Configuration

Subdeployment

Notes

General

Default Delivery

Client

Transactions

Flow Control

Load Balance

Security


Save

Use this page to define the transaction configuration for this JMS connection factory. You can define a transaction manager, create sessions that are JTA user-transaction aware.

Transaction Timeout:

3600

The timeout value (in milliseconds)

 ☐ XA Connection Factory Enabled

Indicates whether a XA queue can be used to create an XAConnection transaction manager. [More Info](#)

Configure Connection Factory: Flow Control



Configuration	Subdeployment	Notes				
General	Default Delivery	Client	Transactions	Flow Control	Load Balance	Security

Flow control allows you to enable a JMS server or destination to slow down message producers when it determines messages thresholds, it instructs producers to limit their message flow (messages per second).

Use this page allows to define the flow control configuration for this JMS connection factory.

Flow Maximum:	<input type="text" value="500"/>	The maximum number of messages will never be allowed to go faster than
Flow Minimum:	<input type="text" value="50"/>	The minimum number of messages producer's flow limit. That is, WebL
Flow Interval:	<input type="text" value="60"/>	The adjustment period of time, in s or vice versa. More Info...
Flow Steps:	<input type="text" value="10"/>	The number of steps used when a p versa. Specifically, the Flow Interva per step). More Info...
<input checked="" type="checkbox"/> Flow Control Enabled		Specifies whether a producer creat JMS server or a destination reaches

- ▶ A *destination* is a lightweight object stored in JNDI.
- ▶ It is the target on a JMS Server for sending or receiving messages.
- ▶ The JMS destination types are:
 - Queue
 - Topic

Create a Queue Destination...



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 dizzyworldJMSModule.

Name:

JNDI Name:

Template:

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

...Create a Queue Destination...

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

The following options are available to target your new JMS destination

Use this page to select the JMS server where you want to target this destination (queue or topic). Optionally, you can click **Advanced Targeting** by which JMS destinations are grouped and targeted to a single JMS server instance. You can reconfigure subdeployment targets later if i

Select a JMS server from the list below and click Finish to target this destination.

Targets:

JMS Servers

☐ dizzyworldJMSServer

☐ dizzyworldJMSServer2

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

...Create a Queue Destination

[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 can create a new subdeployment by clicking the **Create a New Subdeployment** button. You can also reconfigure subdeployment targets.

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

Subdeployments:

[Create a New Subdeployment](#)

What targets do you want to assign to this subdeployment?

Targets:

JMS Servers



dizzyworldJMSServer



dizzyworldJMSServer2

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

Threshold and Quota



- ▶ A *threshold* and a *quota* can be set for Server and Destination objects.
- ▶ A quota is a limit defined for JMS administered objects; it includes these values:
 - The maximum number of bytes that can be stored
 - The maximum number of messages that can be stored
- ▶ A threshold is a limit that triggers message paging, flow control, and logged warnings, using:
 - Upper and lower values for the number of bytes
 - Upper and lower values for the number of messages

Configuring Thresholds and Quotas



Configuration | **Monitoring** | Control | Security | Subdeployment | Notes

General | **Thresholds and Quotas** | Overrides | Logging | Delivery Failure

1

Use this page to define upper and lower byte and/or message thresholds for the queue. Exceeding these thresholds will trigger events such as a maximum size allowed for messages on the queue and select a pre-configured quota, which determines the maximum number of bytes or messages that can be in the queue.

Thresholds

Bytes Threshold High:	<input type="text" value="9223372036854775807"/>	The upper threshold (total number of bytes in this destination) that disables logging and flow control events for the destination. More Info...
Bytes Threshold Low:	<input type="text" value="9223372036854775807"/>	The lower threshold (total number of bytes in this destination) that disables logging and flow control events for the destination. More Info...
Messages Threshold High:	<input type="text" value="9223372036854775807"/>	The upper threshold (total number of messages in this destination) that disables logging and flow control events for the destination. More Info...
Messages Threshold Low:	<input type="text" value="9223372036854775807"/>	The lower threshold (total number of messages in this destination) that disables logging and flow control events for the destination. More Info...

Quotas

Quota:	<input type="text" value="None"/>	A Quota controls the allotment of system resources available to the destination. More Info...
Maximum Message Size:	<input type="text" value="2147483647"/>	The maximum size of a message that is accepted from producer.

2

Section Review



In this section, we learned how to:

- ✓ Identify message-oriented middleware domains (PTP, publish and subscribe)
- ✓ Understand WLS JMS messaging for the PTP and publish and subscribe domains
- ✓ Administer JMS from the console
- ✓ Fine-tune WLS JMS with thresholds and quotas



1. WebLogic Server JMS Administration
2. **Configuring Persistent Messaging**
 - Persistent and Non-Persistent Messages
 - Persistent Backing Stores Using the Console
 - Durable Subscriptions
 - Durable Subscriptions Using the Console
3. Monitoring JMS in WLS

Durable Subscribers and Subscriptions



- ▶ Durable subscribers register durable subscriptions to guarantee message delivery even if subscribers are *inactive*.
- ▶ A subscriber is considered *active* if the Java object that represents it exists.
- ▶ By default, subscribers are non-durable.
- ▶ Administrators configure:
 - Where messages are persisted
 - Persistent connection factories and destinations

When to Use Persistent Messaging

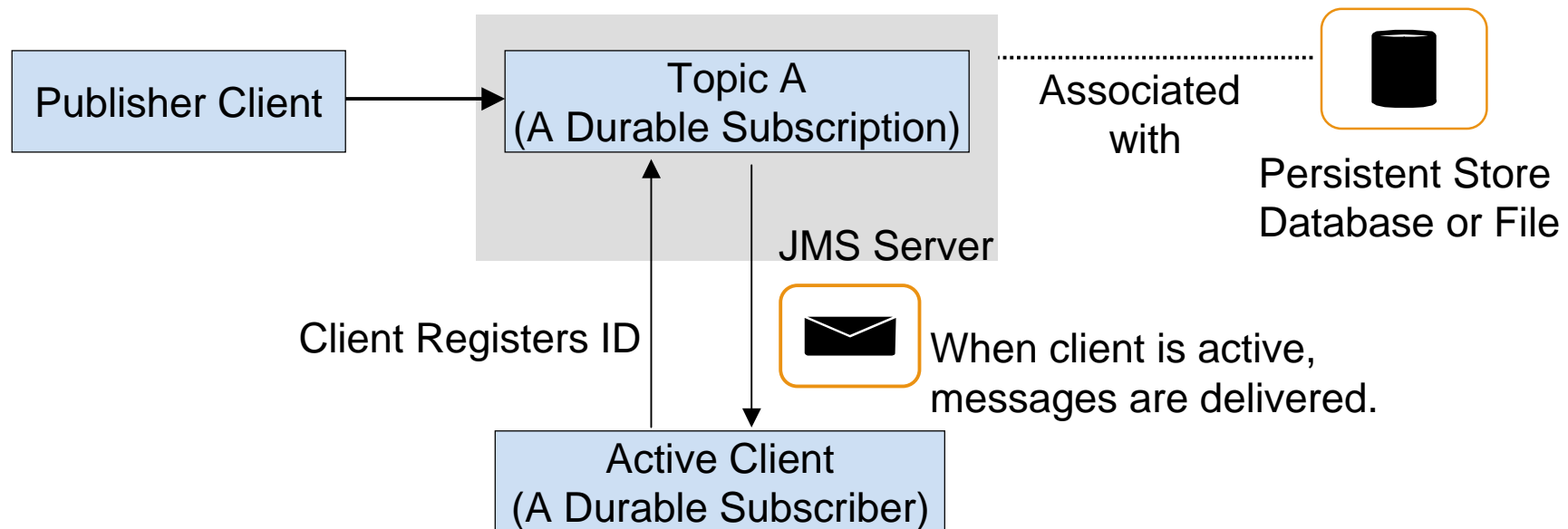


- ▶ Persistent messaging permits messages in memory to be written out to a persistent store.
- ▶ Configure persistent messaging if:
 - Development requires durable subscriptions (use durable subscribers in the application)
 - You require that in-progress messages persist across server restarts

How a Durable Subscription Works



- If a subscriber client is active, messages are delivered normally:



- When the client becomes active again, its ID is used to retrieve and redeliver messages.

Configure a Durable Subscription



- ▶ To configure durable subscriptions, an administrator must:
 - Create and configure a JMS store
 - Configure connection factories or destinations as persistent
 - Associate the JMS store with the JMS Server
- ▶ The JMS store can be configured to use either:
 - A file store
 - A JDBC store (a connection pool)

Create a JMS File Store...



Change Center

View changes and restarts

Pending changes exist. They must be activated to take effect.

Activate Changes

Undo All Changes

Domain Structure

- wl_server
 - Environment
 - Deployments
 - Services
 - Messaging
 - JMS Servers
 - Store-and-Forward Agents
 - JMS Modules
 - Bridges
 - JDBC
 - Persistent Stores**
 - Path Services
 - Foreign JNDI Providers
 - Work Contexts
 - XML Registries
 - XML Entity Caches
 - JCOM
 - Mail Sessions
 - File T3
 - JTA
 - Security Realms

Welcome, weblogic

Connected to: wl_server

Home

Log Out

Preferences

Help

AskBEA

Home > JMS Modules > Summary of Persistent Stores

Summary of Persistent Stores

A persistent store is a physical repository for storing subsystem data, such as persistent JMS messages. It can be either a JDBC-accessible database or a disk-based file. This page summarizes the persistent stores that have been created for this domain.

[Customize this table](#)

Persistent Stores

New

Delete

Showing 1 - 2 of 2

Previous

Next

	Type	Target
Create FileStore	JDBCStore	examplesServer
Create JDBCStore	FileStore	examplesServer
<input type="checkbox"/> WseeFileStore		

New

Delete

Showing 1 - 2 of 2

Previous

Next

Setting up JMS Applications-40

© 2006 BEA Systems, Inc.

436

...Create a JMS File Store



Create a New File Store

File Store Properties

The following properties will be used to identify your new file store.

What would you like to name your new file store?

Name:

Select a server instance for this file store.

Target:

The pathname to the directory on the file system where the file store is kept. This directory must exist on your system, so be sure to create it before

Directory:

Create a JMS JDBC Store...



- ▶ To configure JMS JDBC persistence:
 - Create a JDBC DataSource.
 - Create a JMS store and refer to the JDBC DataSource.
 - Refer to the JMS store from the JMS Server configuration.
- ▶ The required infrastructure (tables, and so on) is created automatically.

...Create a JMS JDBC Store



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

Create a new JDBC Store

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

What would you like to name your new JDBC store?

Name:

Select the server instance for this JDBC store.

Target:

Select the data source for this JDBC store.

Data Source: [Create a New Data Source](#)

The prefix name that is prepended to the JDBC store table names in this JDBC store.

Prefix Name:

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

Assign a Store to a JMS Server



Settings for examplesJMSServer



Configuration [Logging](#) [Targets](#) [Monitoring](#) [Control](#) [Notes](#)

General [Thresholds and Quotas](#) [Session Pools](#)

Save

JMS servers act as management containers for the queues and topics in JMS modules that are targeted to them. A JMS server's primary responsibility for its destinations is to maintain information on what persistent store is used for any persistent messages that arrive on the destinations, and to maintain the states of durable subscribers created on the destinations.

Use this page to define the general configuration parameters for this JMS server.

 Name:	examplesJMSServer	The name of this JMS server. More Info...
 Persistent Store:	<div><div>exampleJDBCStore</div><div>(none)</div><div>WseeFileStore</div><div>exampleJDBCStore</div></div>	The file or database in which this JMS server stores persistent messages. If unspecified, the JMS server uses the default persistent store that is configured on each targeted WebLogic Server instance. More Info...
Paging Directory:		Specifies where message bodies are written when the size of the message bodies in the JMS server exceeds the message buffer size. More Info...
Message Buffer Size:	-1	The amount of memory (in bytes) that this JMS server can use to store message bodies before it writes them to disk. When the JMS server writes the message bodies to disk, it clears them from memory. More Info...

Persistent Connection Factory




Configuration Subdeployment Notes

General **Default Delivery** Client Transactions Flow Control Load Balance Security

Save

Use this page to define the default delivery configuration parameters for this JMS connection factory, such as

Default Priority:	<input type="text" value="4"/>	The default priority used for mess
Default Time-to-Live:	<input type="text" value="0"/>	The maximum length of time, in r 0 indicates that the message has
Default Time-to-Deliver:	<input type="text" value="0"/>	The delay time, in milliseconds, b
Default Delivery Mode:	<div>Persistent Persistent Non-Persistent 2147483647</div>	The default delivery mode used fo
Default Compression Threshold:		The number of bytes for the seria received by the JMS message pro
Send Timeout:	<input type="text" value="10"/>	The maximum length of time, in r the message being sent. More Inf
 Default Unit-of-Order for Producer:	<div>None</div>	The default Unit-of-Order name fr order, even among multiple recip

Configure a Persistent Destination



Configuration | **Monitoring** | Control | Security | Subdeployment | Notes

General | Thresholds and Quotas | **Overrides** | Logging | Delivery Failure

Destinations can override some of the settings (such as priority) that a message producer includes with its messages. Use this p

Priority Override:	<input type="text" value="-1"/>	The priority assigned to all messages that arrive at the destination. A value of -1 specifies that the destination will not override the Priority assigned by the message producer.
Time-to-Live Override:	<input type="text" value="-1"/>	The time-to-live assigned to all messages that arrive at the destination. A value of -1 specifies that this setting will not override the TimeToLive specified by the message producer.
Time-to-Deliver Override:	<input type="text" value="-1"/>	The default delay, either in milliseconds or as a schedule, before the message is delivered to the destination. A value of -1 specifies that the destination will use the TimeToDeliver setting specified by the message producer. More Info...
Delivery Mode Override:	<div><div>Persistent</div><div>No-Delivery</div><div>Persistent</div><div>Non-Persistent</div></div>	The delivery mode assigned to all messages that arrive at the destination. The default is Persistent.

Section Review



In this section, we learned how to:

- ✓ Distinguish between persistent and non-persistent messages
- ✓ Configure persistent backing stores using the console
- ✓ Manage durable subscriptions



Road Map



1. WebLogic Server JMS Administration
2. Configuring Persistent Messaging
3. **Monitoring JMS in WLS**
 - Using the Administration Console to Track JMS Statistics

Statistics for JMS Objects



- ▶ Statistics are provided for the following JMS objects:
 - JMS servers
 - Connections
 - Destinations

Monitor JMS Servers...



Change Center
[View changes and restarts](#)
Pending changes exist. They must be activated to take effect.
Activate Changes
Undo All Changes

Domain Structure
wl_server
├── Environment
├── Deployments
├── Services
│ ├── Messaging
│ │ ├── **JMS Servers**
│ │ ├── Store-and-Forward Agents
│ │ └── JMS Modules
│ ├── Bridges
│ ├── JDBC
│ ├── Persistent Stores
│ ├── Path Services
│ ├── Foreign JNDI Providers
│ ├── Work Contexts
│ ├── XML Registries
│ ├── XML Entity Caches
│ ├── jCOM
│ ├── Mail Sessions
│ └── File T3

Welcome, weblogic Connected to: **wl_server** [Home](#) [Log Out](#) [Preferences](#) [Help](#) [AskBEA](#)
Home > examples-jms > exampleTopic > JMS Modules > examples-jms > jms/MULTIDATASOURCE_MDB_QUEUE > Summary of JMS Servers > examplesJMSServer > Summary of JMS Servers > examplesJMSServer > **Summary of JMS Servers**
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
New Delete Showing 1 - 2 of 2 Previous | Next

<input type="checkbox"/>	Name	Persistent Store	TargetName
<input type="checkbox"/>	examplesJMSServer	exampleJDBCStore	examplesServer
<input type="checkbox"/>	WseeJMSServer	WseeFileStore	examplesServer

New Delete Showing 1 - 2 of 2 Previous | Next

...Monitor JMS Servers



Settings for examplesJMSServer

[Configuration](#) [Logging](#) [Targets](#) [Monitoring](#) [Control](#) [Notes](#)

[Monitoring](#) [Active Destinations](#) [Active Transactions](#) [Active Connections](#) [Active Session Pools](#)

This page allows you to view active connections to this JMS server.

[Customize this table](#)

Connections

[Destroy](#)

Showing 1 - 1 of 1 [Previous](#) | [Next](#)

<input type="checkbox"/>	Client ID	Sessions Current Count	Sessions High Count	Sessions Total Count	Host Address
<input type="checkbox"/>		16	16	17	127.0.0.1

[Destroy](#)

Showing 1 - 1 of 1 [Previous](#) | [Next](#)

Monitor Destinations



Settings for examplesServer.jms

Configuration Logging Targets Monitoring Control Notes

Monitoring Active Destinations Active Transactions Active Connections Active Session Pools

This page allows you to view active destinations targeted to this JMS server.

[Customize this table](#)

Destinations

Production Consumption Insertion

Showing 1 - 4 of 4 Previous | Next

<input type="checkbox"/>	Name	Messages Current	Messages Pending	Messages High	Messages Received	Messages Threshold	DestinationType	State	Production Paused	Insertion Paused	Consumption Paused
<input type="checkbox"/>	examples-jms!exampleQueue	0	0	0	0	0	0	advertised_in_local_jndi	false	false	false
<input type="checkbox"/>	examples-jms!exampleTopic	0	0	0	0	0	0	advertised_in_local_jndi	false	false	false
<input type="checkbox"/>	examples-jms!jms/MULTIDATASOURCE_MDB_QUEUE	0	0	0	0	0	0	advertised_in_local_jndi	false	false	false
<input type="checkbox"/>	examples-jms!quotes	0	0	0	0	0	0	advertised_in_cluster_jndi	false	false	false

Production Consumption Insertion

Showing 1 - 4 of 4 Previous | Next

Section Review



In this section, we learned how to:

- ✓ Use the administration console to display JMS statistics



Configuring JMS Servers and Destinations

- ▶ For details on the exercise, refer to the Lab Guide.
- ▶ If questions arise, ask the instructor.
- ▶ The instructor will determine the stop time.



Module Review



In this section, we learned how to:

- ✓ Understand messaging concepts
- ✓ Understand WebLogic Server's JMS support
- ✓ Configure JMS servers, queues, and topics
- ✓ Monitor JMS servers, queues, and topics

