

13

Understanding JDBC and Configuring Data Sources

Objectives

After completing this lesson, you should be able to:

- Configure JDBC and JDBC data sources
- Configure data source scope
- Contrast two-tier and multi-tier JDBC architecture
- Configure a connection pool
- List the benefits of connection pools
- Describe how data sources are used
- Deploy JDBC resources to a target
- View the server JNDI tree
- Complete a connection pool checklist
- Explain the components of JDBC URLs
- Monitor and test a data source

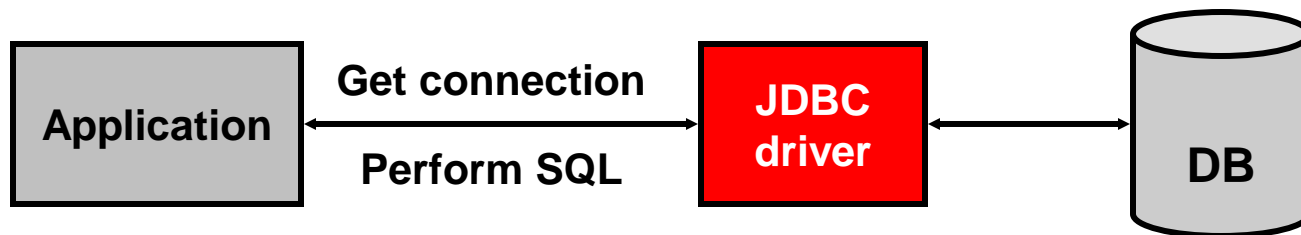
Road Map

- Overview of JDBC
 - High-level architecture of JDBC and the driver model
 - Design of a multi-tier architecture
 - Drivers provided by Oracle WebLogic Server
- Data sources
- Monitoring and testing data sources



JDBC Review

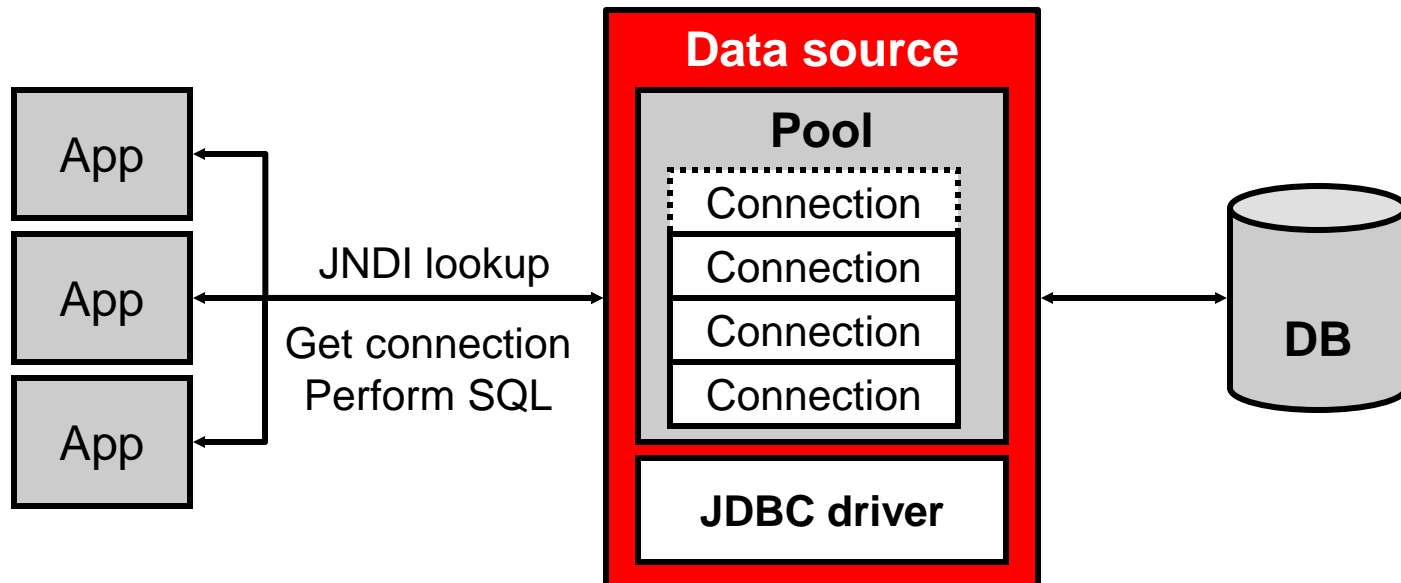
- The Java Database Connectivity (JDBC) specification:
 - Is a platform- and vendor-independent mechanism for accessing and updating a database
 - Provides transparency from proprietary vendor issues
 - Requires the use of a *driver*
- JDBC drivers are supplied by WebLogic Server or by your database vendor.



JDBC Data Sources

Data sources:

- Enable database connectivity to be managed by the application server
- Are obtained by applications from the server's JNDI tree
- Use a dynamic pool of reusable database connections

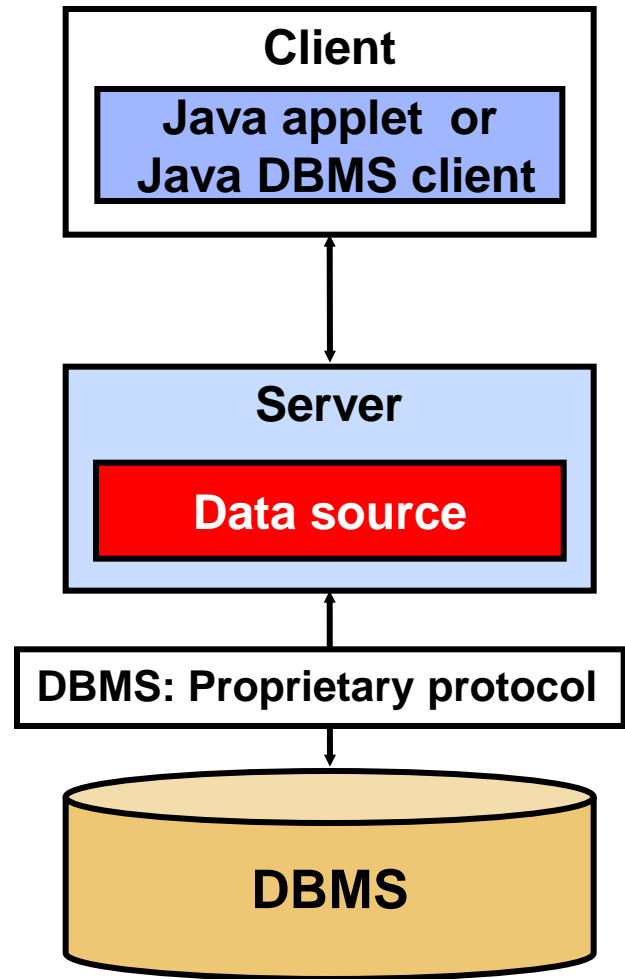


Data Source Scope

- Each data source configuration or “module” is persisted as a separate XML document.
- The system modules that are created with the console or WLST are:
 - Stored in the domain’s `config/jdbc` directory
 - Available to all applications in the domain
- Application-specific modules are:
 - Deployed as part of Java Platform, Enterprise Edition (Java EE) enterprise applications
 - Accessible only by the containing application

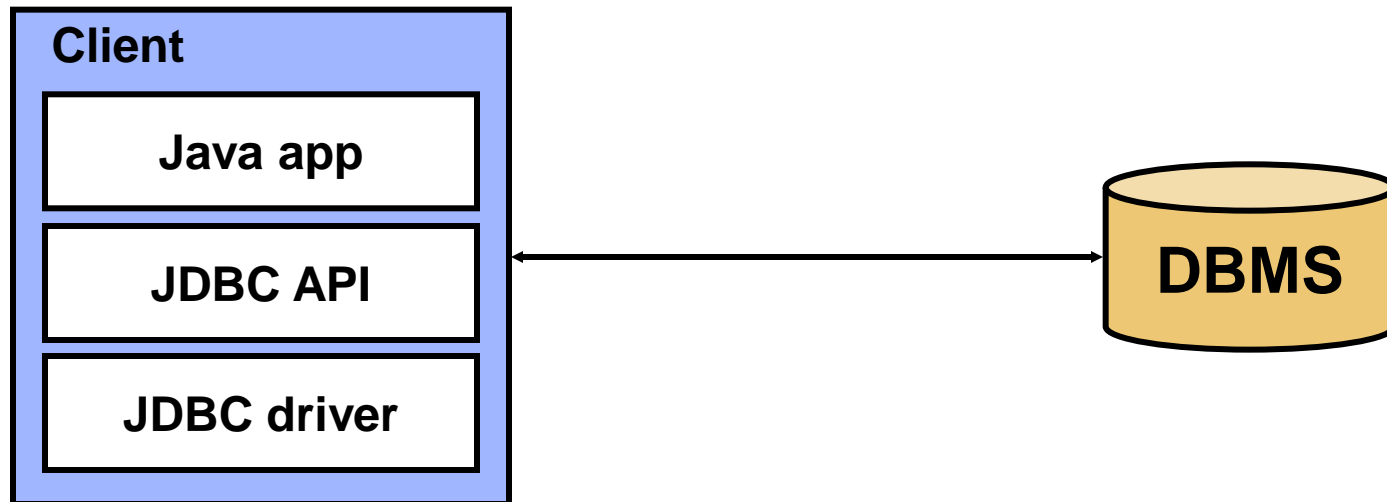
Multi-Tier Architecture

- In the multi-tier model, commands are sent to a “middle tier” of services, which then sends the commands to the DBMS.
- The DBMS processes the commands and sends the results back to the middle tier, which then sends them to the client.



Type 4 Drivers

Type 4 drivers are “all-Java” driver implementations that do not require client-side configuration.



WebLogic JDBC Drivers

- Oracle and third-party drivers are included in the WLS installation for many popular database products:
 - Oracle 9*i*, 10*g*, and 11*g*
 - Sybase Adaptive Server
 - Microsoft SQL Server
 - IBM DB2
 - Informix
 - MySQL
 - PointBase
- By default, these drivers are added to the server's classpath.

Road Map

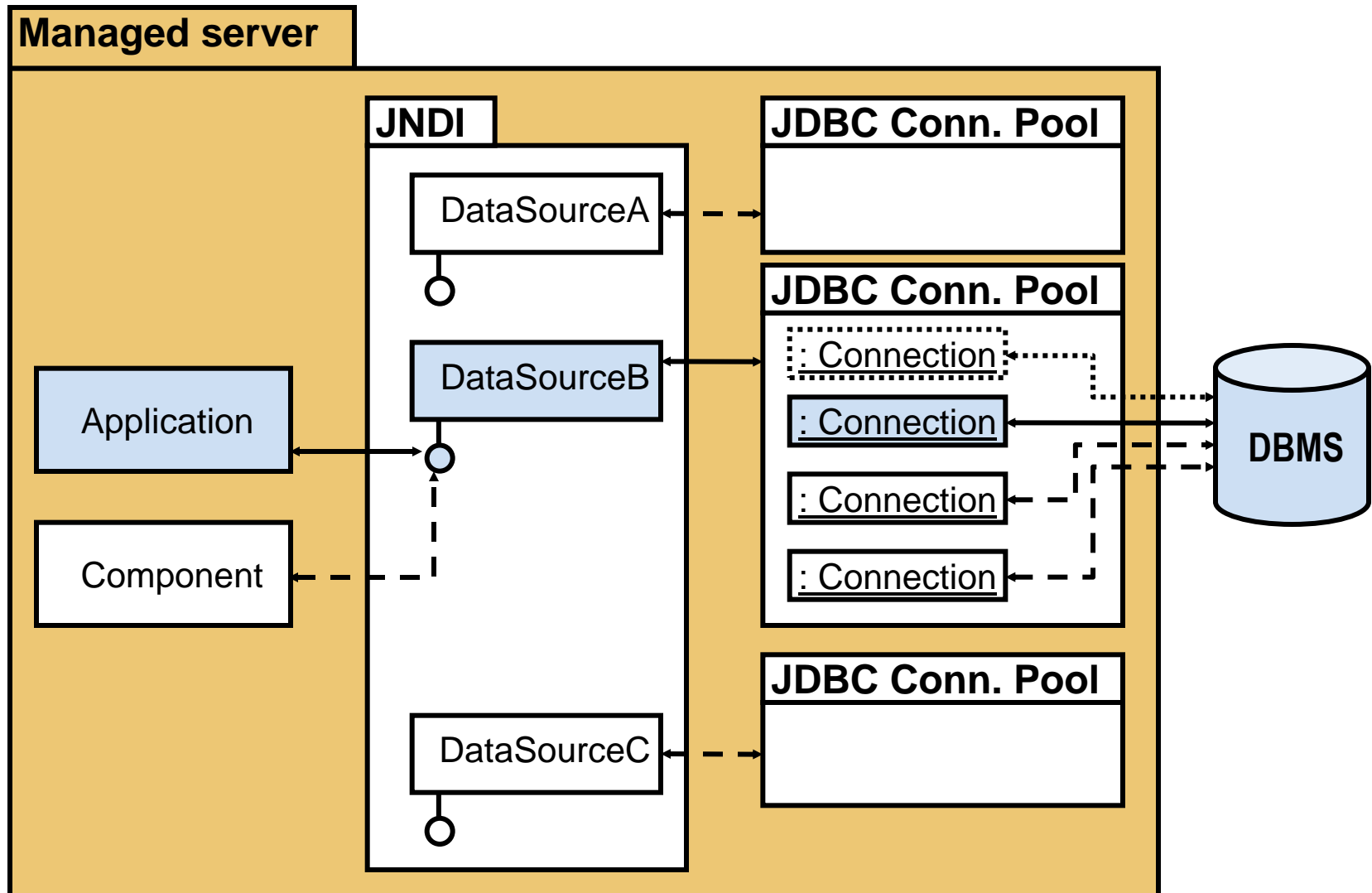
- Overview of JDBC
- Data sources
 - Describing a data source and how it works
 - Using the Administration Console to create a data source
- Monitoring and testing data sources



What Is a Connection Pool?

- A connection pool is a group of ready-to-use database connections associated with a data source.
- Connection pools:
 - Are created at Oracle WebLogic Server startup
 - Can be administered using the Administration Console
 - Can be dynamically resized to accommodate increasing or decreasing load

JDBC Connection Pooling



Benefits of Connection Pools

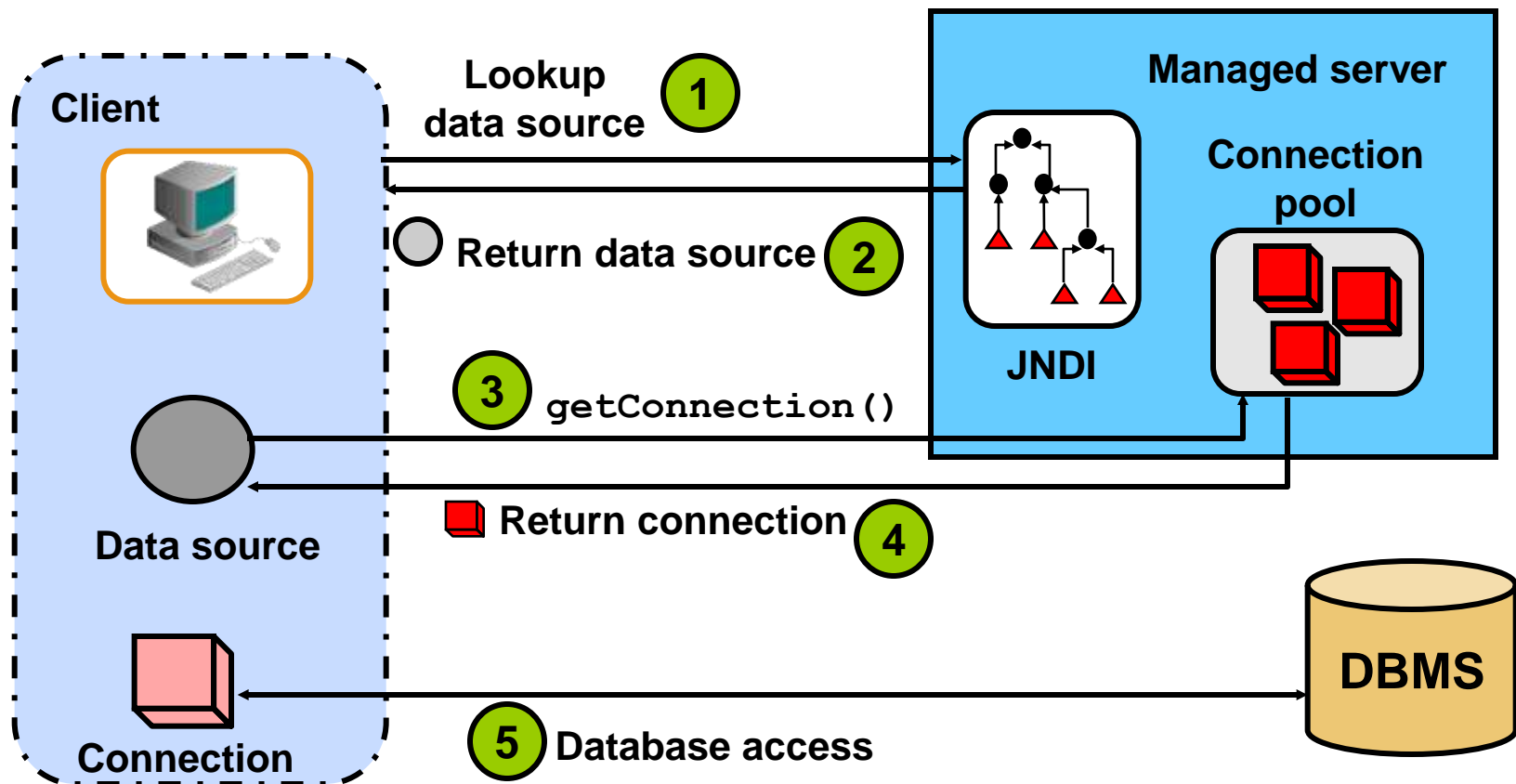
- The following are some advantages of connection pooling:
 - Connection time and overhead are saved by using an existing database connection.
 - It facilitates easier management because connection information is managed in one location.
 - The number of connections to a database can be controlled.
 - The DBMS can be changed without the application developer having to modify the underlying code.
- A connection pool allows an application to “borrow” a DBMS connection.

Modular Configuration and Deployment of JDBC Resources

- The JDBC configurations in WebLogic Server are stored in XML documents:
 - All JDBC configurations must conform to the new `weblogic-jdbc.xsd` schema.
 - IDEs and other tools can validate the JDBC modules based on the schema.
- You create and manage JDBC resources either as system modules or as application modules.
- The JDBC application modules are a WLS-specific extension of Java EE modules and can be deployed either within a Java EE application or as stand-alone modules.

How Data Source Connection Pools Are Used

A client retrieves a data source through a JNDI lookup and uses it to obtain a database connection.



Creating a Data Source Using the Administration Console

Domain Structure

- MedRecDomain
 - Environment
 - Deployments
 - Services
 - Messaging
 - JDBC
 - Data Sources**
 - Multi Data Sources
 - Data Source Factories
 - Persistent Stores
 - Foreign JNDI Providers
 - Work Contexts
 - XML Registries
 - XML Entity Caches

Summary of JDBC Data Sources

Data Sources(Filtered - More Columns Exist)

Showing 1 to 1 of 1 Previous | Next

<input type="checkbox"/>	Name	JNDI Name	Targets
<input type="checkbox"/>	MedRecGlobalDataSourceXA	jdbc/MedRecGlobalDataSourceXA	MedRecClust1

Create a New JDBC Data Source

Back Next Finish Cancel

JDBC Data Source Properties

What would you like to name your new JDBC data source?

*** Name:**

What JNDI name would you like to assign to your new JDBC Data Source?

JNDI Name:

What database type would you like to select?

Database Type: Oracle

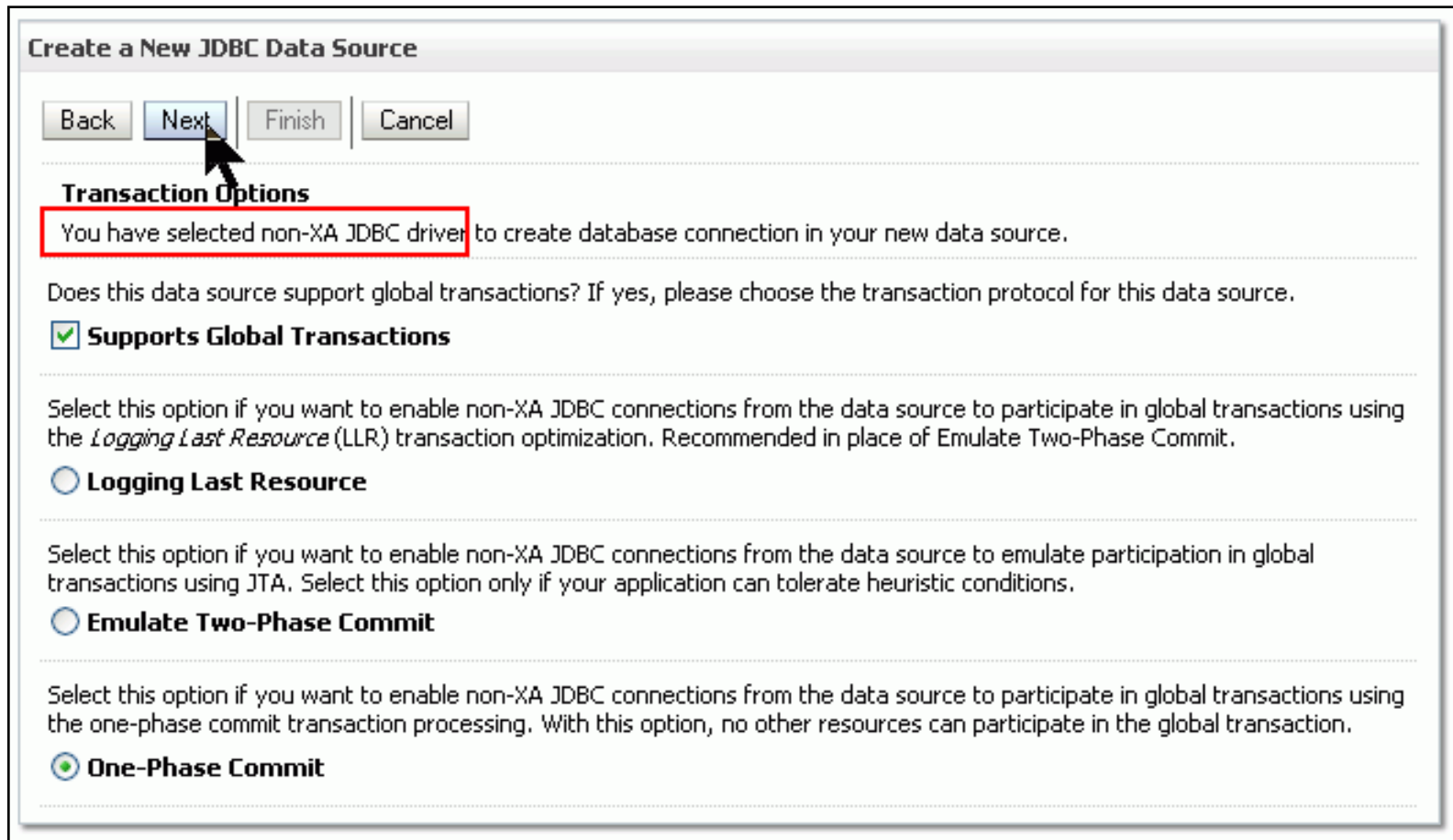
What database driver would you like to use to create database connections?

Database Driver: *Oracle's Driver (Thin) for Service connections; Versions:9.0.1,9.2.0,10,11

Notice non-XA

Non-XA Configuration

This appears only if a non-XA driver was selected previously.



The screenshot shows a Java Swing dialog box titled "Create a New JDBC Data Source". At the top, there are four buttons: "Back", "Next", "Finish", and "Cancel". The "Next" button is highlighted with a mouse cursor. Below the buttons, the "Transaction Options" section is enclosed in a red rectangular box. The text inside this box reads: "You have selected non-XA JDBC driver to create database connection in your new data source." Below this, a question is posed: "Does this data source support global transactions? If yes, please choose the transaction protocol for this data source." There are three radio button options: "Supports Global Transactions" (which is selected), "Logging Last Resource", and "Emulate Two-Phase Commit". Below these, there is a fourth option, "One-Phase Commit", which is also selected. Each option has a descriptive paragraph explaining its use.

Create a New JDBC Data Source

Back Next Finish Cancel

Transaction Options

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

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

☒ **Supports Global Transactions**

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

☐ **Logging Last Resource**

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

☐ **Emulate Two-Phase Commit**

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

☒ **One-Phase Commit**

Data Source Connection Properties

Create a New JDBC Data Source

Back Next Finish Cancel

Connection Properties

Define Connection Properties.

What is the name of database you would like to connect to?

Database Name: orcl.us.oracle.com

What is the name or IP address of the database server?

Host Name: localhost

What is the port on the database server used to connect to the database?

Port: 1521

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

Database User Name: HR

What is the database account password to use to create database connections?

Password: ●●

Confirm Password: ●●

Not just "name." Be mindful of whether you are entering a *service* name vs. *instance* name vs. *database* name.

Sample schemas

Test Configuration

Create a New JDBC Data Source

Test Configuration Back Next Finish Cancel

Test Database Connection
Test the database availability and the connection properties you provided.

What is the full package name of JDBC driver class used to create database connections in the connection pool?
Driver Class Name:

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

What database account user name do you want to use to create database connections?
Database User Name:

What is the database account password to use to create database connections?
Password:

Confirm Password:

What are the properties to pass to the JDBC driver when creating database connections?
Properties:

What table name or SQL statement would you like to use to test database connections?
Test Table Name:

Create a New JDBC Data Source

Back Next **Finish** Cancel

Select Targets

Servers

☐ MedRecAdmSvr

☒ MedRecSvr3

Clusters

☐ MedRecClust1

☐ All servers in the cluster

☐ Part of the cluster

☐ MedRecSvr2

☐ MedRecSvr1

Back Next Finish Cancel

Connection Pool Configuration

Summary of JDBC Data Sources

Data Sources(Filtered - More Columns Exist)

New Delete Showing 1 to 2 of 2 Previous Next

<input type="checkbox"/>	Name	JNDI Name	Targets
<input type="checkbox"/>	MedRecGlobalDataSourceXA	jdbc/MedRecG	
<input type="checkbox"/>	MedRecHR	MedRecHR	

New Delete

Settings for MedRecHR

Configuration Targets Monitoring Control Security Notes

General **Connection Pool** Transaction Diagnostics Identity Options

Save

URL: jdbc:oracle:thin:@localhost:1521/orcl.us.oracle.com





Driver Class Name: oracle.jdbc.OracleDriver

Properties: user=HR

Password:

Confirm Password:

Connection Pool Advanced

Initial Capacity:	<input type="text" value="1"/>	Connection pool size
Maximum Capacity:	<input type="text" value="15"/>	
Capacity Increment:	<input type="text" value="1"/>	Grows pool when more connections are needed
 Statement Cache Type:	<input type="text" value="LRU"/> 	Periodically tests for bad connections and closes
Statement Cache Size:	<input type="text" value="10"/>	
Advanced		
<input type="checkbox"/> Test Connections On Reserve		Periodically closes idle connections
Test Frequency:	<input type="text" value="120"/>	
Test Table Name:	<input type="text" value="SQL SELECT 1 FROM DUAL"/>	More options not shown
Seconds to Trust an Idle Pool Connection:	<input type="text" value="10"/>	
Shrink Frequency:	<input type="text" value="900"/>	
 Init SQL:	<input type="text"/>	
Connection Creation Retry Frequency:	<input type="text" value="0"/>	
 Login Delay:	<input type="text" value="0"/>	

Targeting a Data Source

Deploy data sources to one or more servers in your domain.

Settings for MedRecHR

Configuration **Targets** Monitoring Control Security Notes

Save

Servers

☐ MedRecAdmSvr

☒ MedRecSvr3

Clusters

☐ MedRecClust1

- ☐ All servers in the cluster
- ☐ Part of the cluster
 - ☐ MedRecSvr2
 - ☐ MedRecSvr1

Save

Viewing the Server JNDI Tree via the Administration Console

The screenshot illustrates the steps to view the JNDI tree for a specific server in the Oracle Administration Console. It is divided into three main sections:

- Domain Structure:** A tree view on the left showing the hierarchy of the domain. The 'Servers' node under 'Environment' is highlighted with a red circle and the number 1.
- Settings for MedRecSvr3:** A panel on the right showing the configuration for the selected server. The 'General' tab is active, and the 'View JNDI Tree' link is highlighted with a red circle and the number 2. A yellow callout box points to this link with the text: "Confirm data source deployment using the server's JNDI tree."
- JNDI Tree Structure:** A new panel showing the JNDI tree for MedRecSvr3. The 'MedRecHR' node is highlighted with a red circle and the number 3.

The 'Settings for MedRecHR' panel on the right shows the following details:

Settings for MedRecHR	
Overview	
Binding Name:	MedRecHR
Class:	weblogic.jdbc.common.internal.RmiDataSource_1031_WLStub
Hash Code:	33907648
toString Results:	ClusterableRemoteRef(-37818975089890092085:localhost: [7025,7025,-1,-1,-1,-1]:MedRecDomain:MedRecSvr3 [-37818975089890092085:localhost: [7025,7025,-1,-1,-1,-1]:MedRecDomain:MedRecSvr3/283])/283

Listing the JNDI Contents via WLST

- WLST provides a command-line utility for viewing the JNDI bindings.
- `jndi()` changes to the JNDI tree and `ls()` lists the bindings.

```
wls:/offline> connect("weblogic","welcome1","t3://localhost:7020")
```

```
wls:/base_domain/serverConfig> jndi()
```

```
wls:/base_domain/jndi> cd('AdminServer')
```

```
wls:/base_domain/jndi/AdminServer> ls()
```

dr--	ejb	
dr--	javax	
dr--	weblogic	
-r--	cgDataSource	weblogic.rmi.cluster.ClusterableRemoteObject
-r--	cgDataSource-nonXA	weblogic.rmi.cluster.ClusterableRemoteObject
-r--	mejbmejb_jarMejb_EO	weblogic.rmi.cluster.ClusterableRemoteObject
-r--	samplesDataSource	weblogic.rmi.cluster.ClusterableRemoteObject

JDBC data
source

Demonstration

- Configure data sources for Oracle Database.
- Go to OTN > Tutorials > Fusion Middleware > Oracle WebLogic Server 10.3 > Deploy J2EE Applications > [Configure Data Sources.](#)

JDBC URLs

Database locations are specified using a JDBC Uniform Resource Locator (URL).

- Example 1:
 - This URL specifies that the `oracle:thin` subprotocol should be used to connect to an Oracle Database:

```
jdbc:oracle:thin:@dbhost:1521:SALESINFO
```

- Example 2:
 - This URL can be used to access a PointBase database:

```
jdbc:pointbase:server://dbhost:9092/HRDATABASE
```

Connection Properties

- Are key/value pairs
- Are used to configure JDBC connections
- Are passed to the driver during connection setup

Specifying Connection Properties

A partial list of connection properties for the supplied drivers:

Driver	Some Connection Properties
Oracle	User, Password, ServerName, ServiceName, PortNumber
Sybase	User, Password, ServerName, DatabaseName, PortNumber
MSSQL	User, Password, ServerName, DatabaseName, PortNumber
Informix	User, Password, ServerName, DatabaseName, PortNumber
PointBase	cache.size, crypto.communication, database.home, database.pagesize

Road Map

- Overview of JDBC
- Data sources
- Monitoring and testing data sources
 - Monitoring
 - Testing
 - Suspend/resume



Monitoring and Testing a Data Source

Messages

✓ Test of MedRecHR on server MedRecSvr3 was successful.

Settings for MedRecHR

Configuration Targets **Monitoring** Control Security Notes

Statistics **Testing**

Test Data Source(Filtered - More Columns Exist)

Test Data Source

Server

MedRecSvr3

Test Data Source

Settings for MedRecHR

Configuration Targets **Monitoring** Control Security Notes

Statistics Testing

Customize this table

Deployed Instances of this Data Source(Filtered - More Columns Exist)

Showing 1 to 1 of 1 Previous | Next

Server	State	Connections Total Count	Current Capacity	Waiting For Connection High Count	Highest Num Available	Active Connections Average Count
MedRecSvr3	Running	1	1	0	1	0

Showing 1 to 1 of 1 Previous | Next

Monitor data source statistics.

Data source retested successfully.

Connection Pool Life Cycle

For this data source...

Settings for testSample

Configuration Targets Monitoring **Control** Security Notes

Use this page to manually control each instance of this JDBC data source.

▶ [Customize this table](#)

Deployed Instances of this Data Source

Shrink Reset Clear Statement Cache Showing 1 to 1 of 1 Previous | Next

Suspend ▼ Resume Shutdown ▼ Start

<input checked="" type="checkbox"/>	Server Name ^	State	Status of Last Action
<input checked="" type="checkbox"/>	MedRecSvr1	Running	None

Shrink Reset Clear Statement Cache Showing 1 to 1 of 1 Previous | Next

Suspend ▼ Resume Shutdown ▼ Start

Suspend
Force Suspend

...on a given server...

...take this action.

Quiz

Which of the following is NOT an available configuration attribute for a JDBC data source?

1. Host name
2. Queue size
3. Test frequency
4. Initial capacity
5. Capacity increment

Quiz

Which are the two levels of data sources available in Oracle WebLogic Server?

1. Connection
2. Web
3. Application
4. Process
5. System

Quiz

Client applications look up data sources from the local server's _____ tree:

1. Application
2. Web
3. LDAP directory
4. JNDI
5. System

Summary

In this lesson, you should have learned how to:

- Define JDBC high-level architecture
- Configure Oracle WebLogic Server–provided JDBC driver types
- Create data source definitions
- Create connection pool definitions
- Manage JDBC resources using the Administration Console

Practice 13 Overview: Configuring JDBC Data Sources

This practice covers the following topics:

- Creating JDBC modules (via GUI and WLST)
- Deploying JDBC modules
- Testing JDBC modules