Setting Up JDBC

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

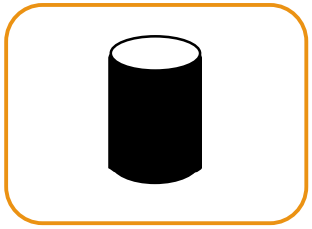
- ✓ Describe the high level architecture of JDBC
- ✓ List the four driver types and those provide by WLS
- ✓ Describe and configure Data Sources
- ✓ Use the Administration Console to manage JDBC resources

Road Map

1. **Overview of JDBC**
   - High Level Architecture of JDBC and the Driver Model
   - Four Different Driver Types
   - Differences Between Two-tier and Multi-tier Models
   - Drivers Provided by WebLogic Server
2. Data Sources
3. Monitoring and Testing Data Sources

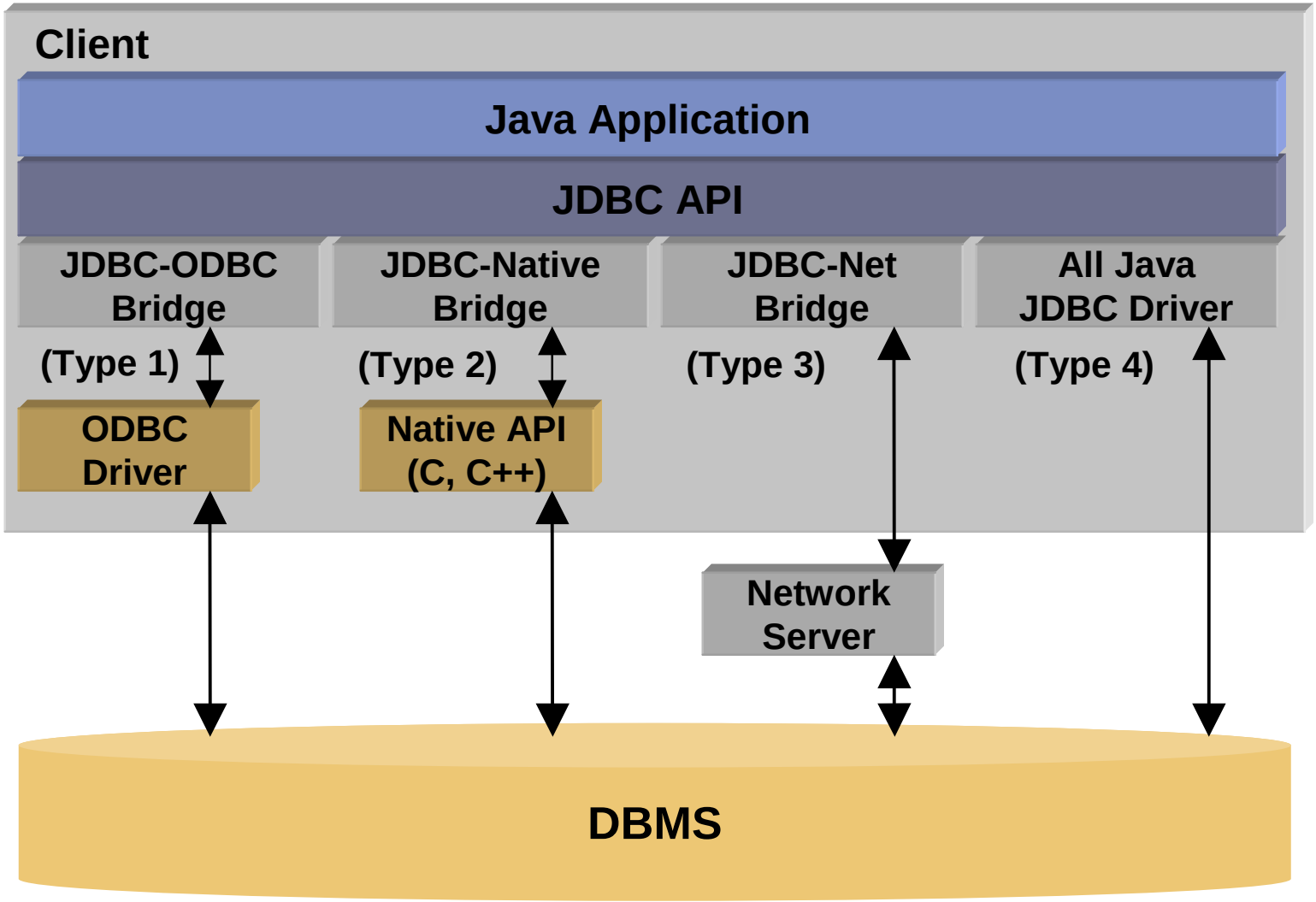
What Is JDBC?

- JDBC is an API for accessing databases in a uniform way.
- JDBC provides:
  - Platform independent access to databases
  - Location transparency
  - Transparency to proprietary database issues
  - Support for both two-tier and multi-tier models for database access



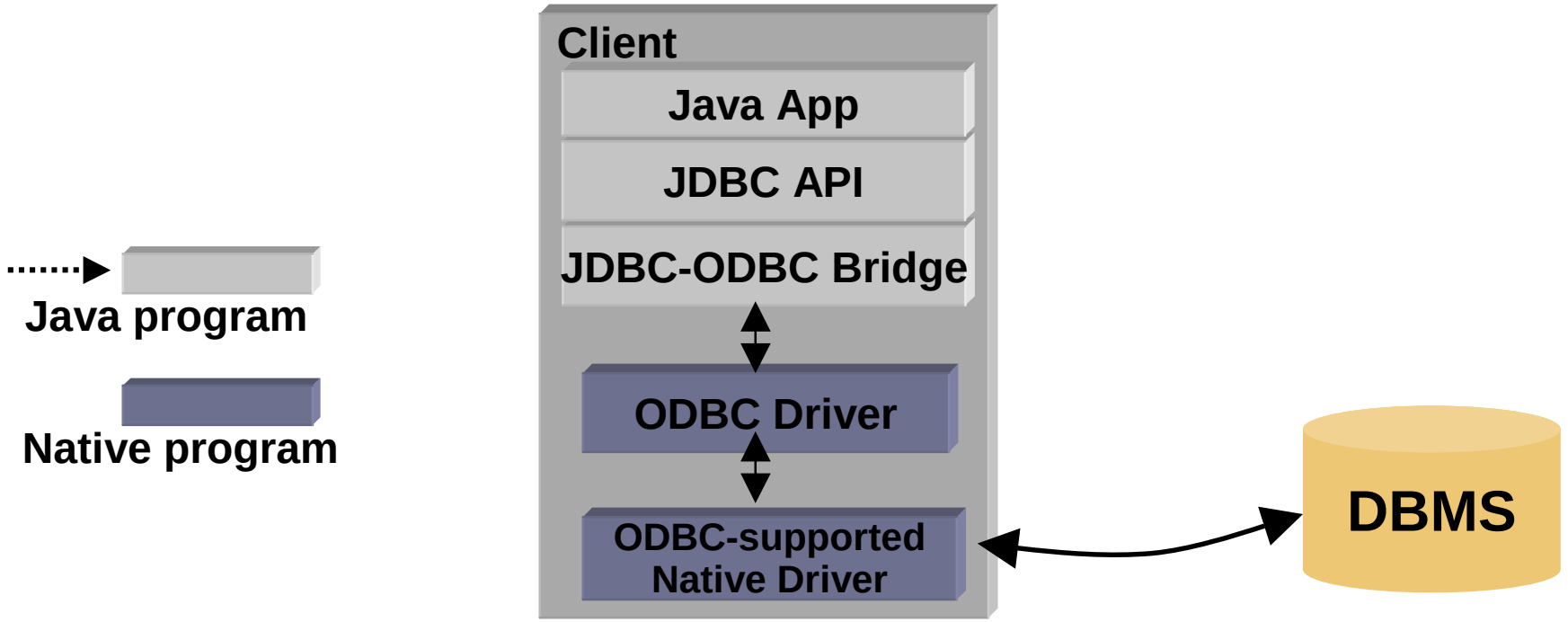
Data

JDBC Architecture



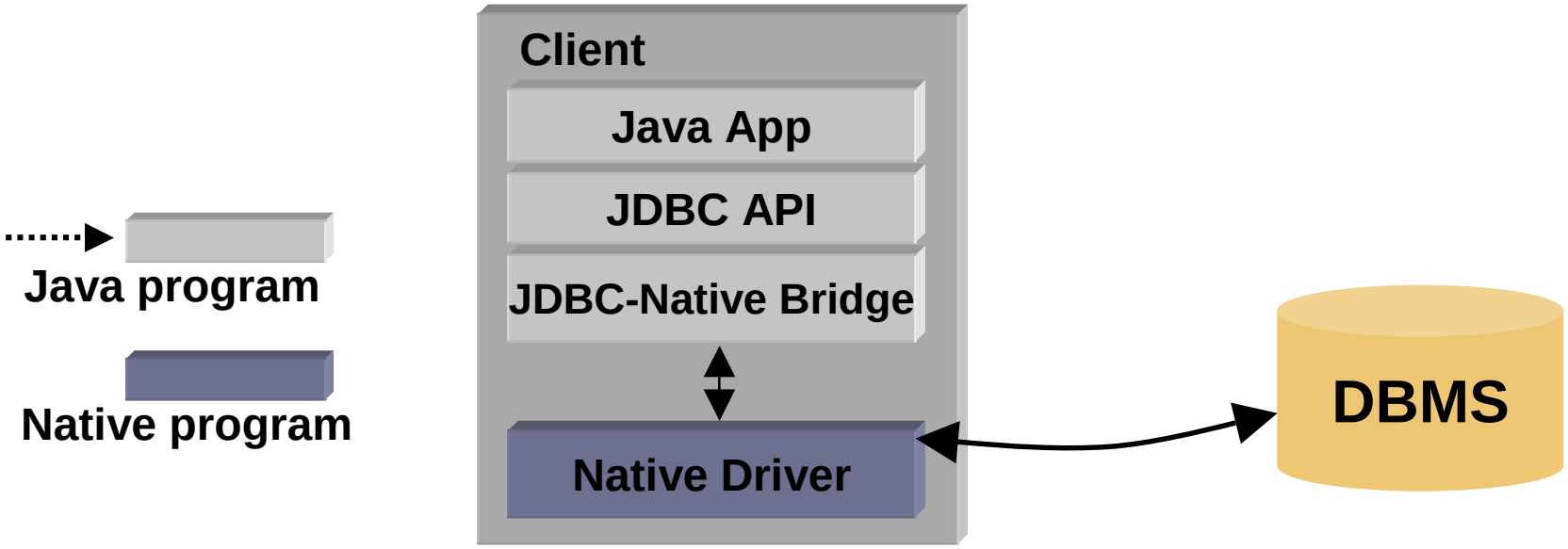
Type 1 Driver

- The Type 1 driver:
  - Is a JDBC-ODBC bridge
  - Usually runs on Windows
  - Requires ODBC driver to be installed on client machine

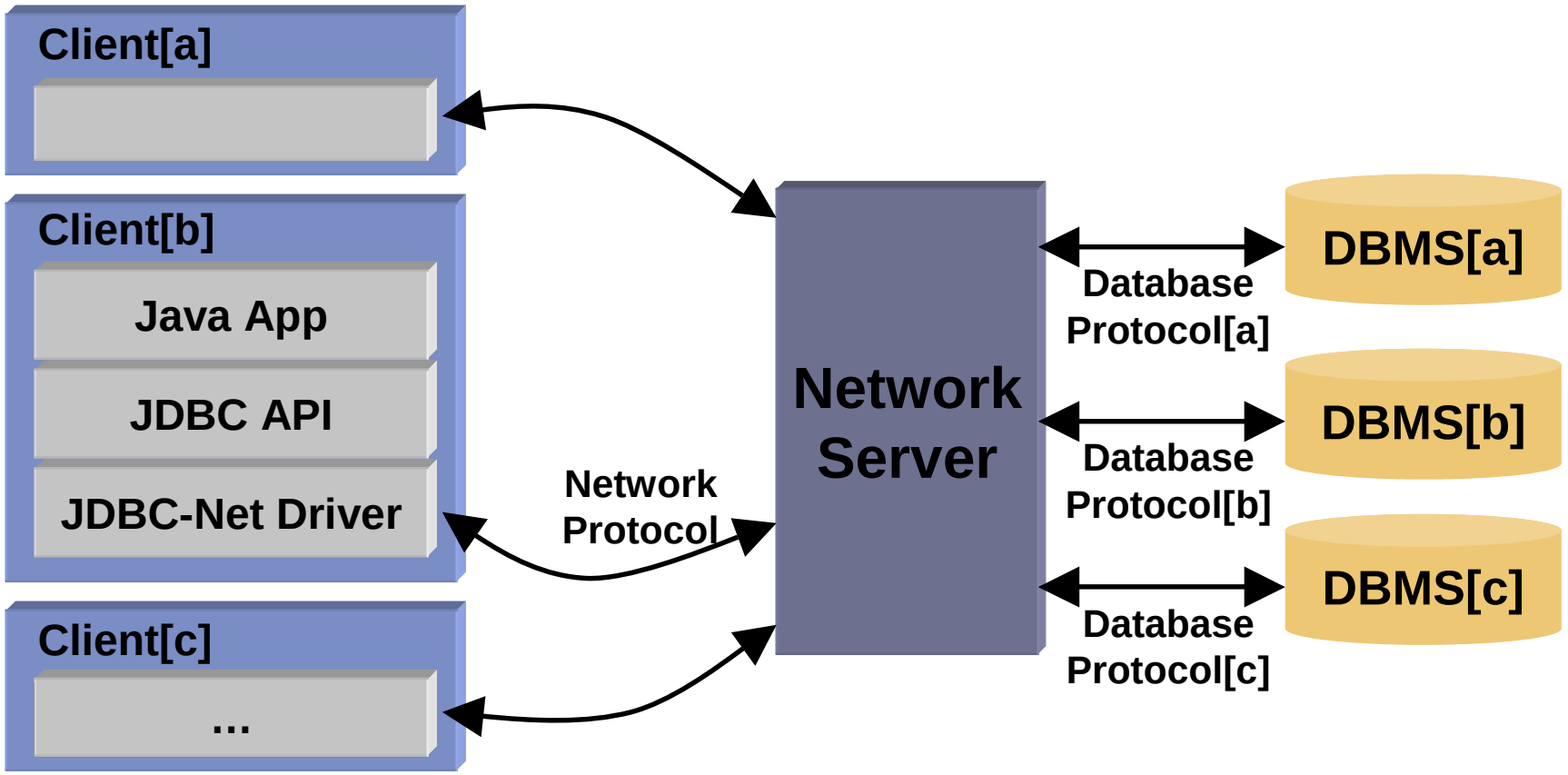


Type 2 Drivers

- The Type 2 driver:
  - Requires a native driver to be already installed on the client machine
  - The driver converts JDBC calls to native API calls of the database

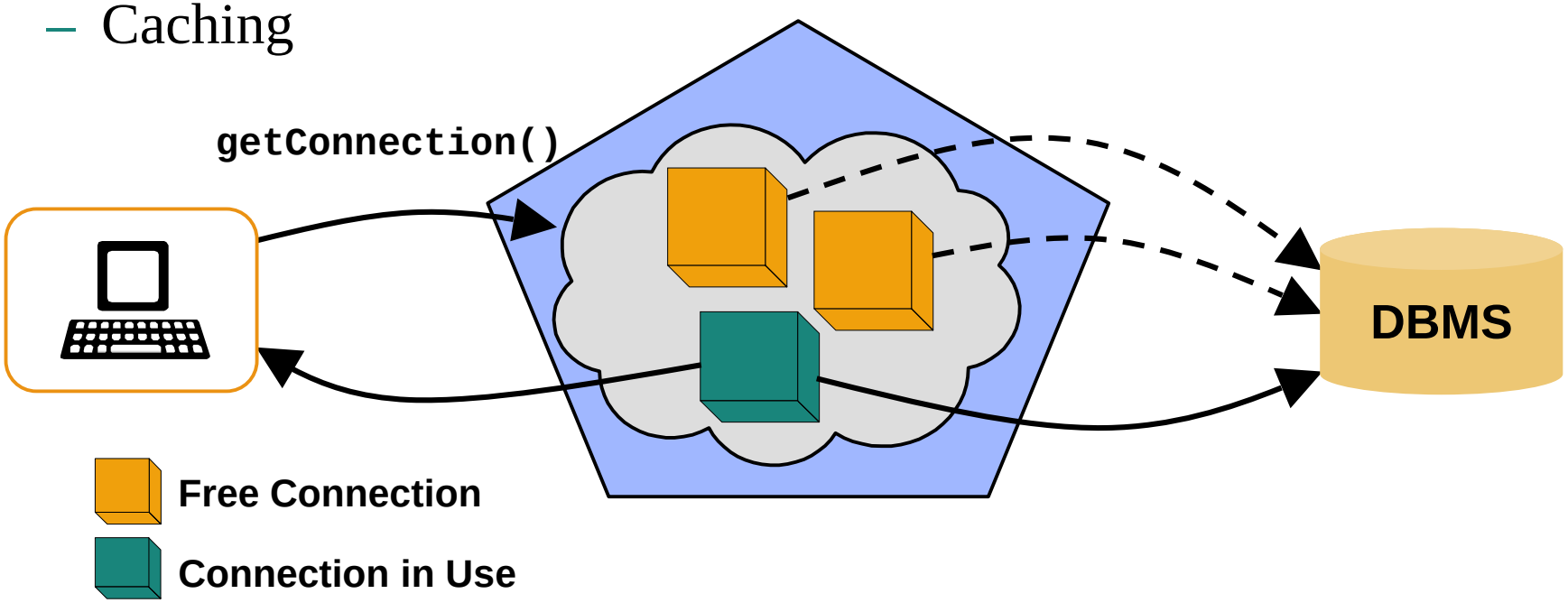


Type 3 Drivers…



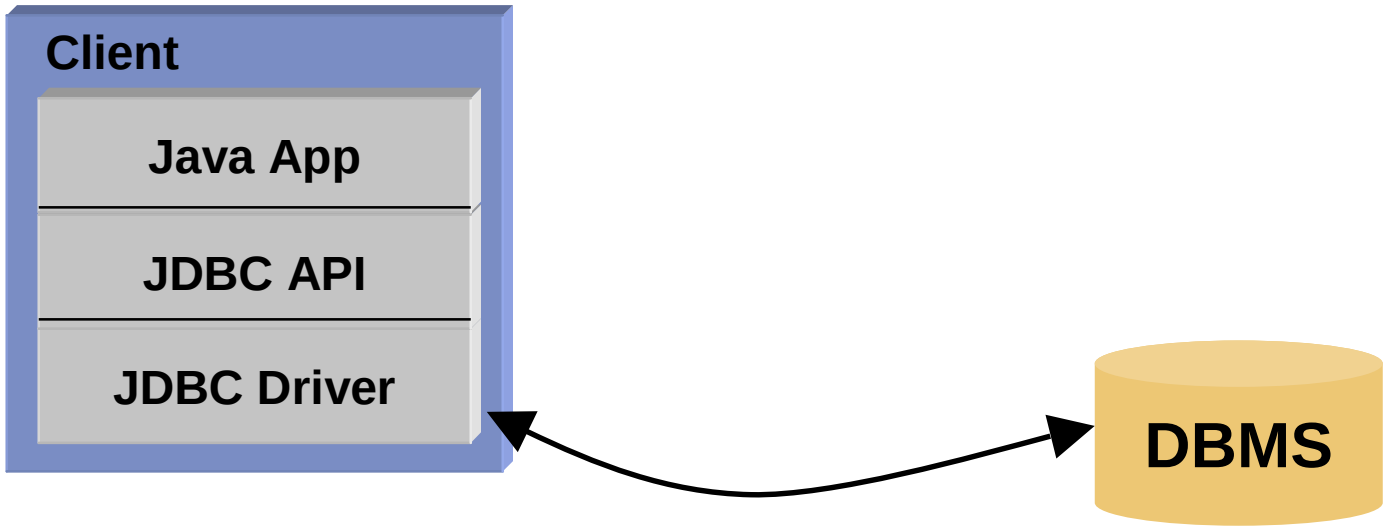
...Type 3 Drivers

- A network server can apply several techniques to boost performance:
  - Connection pooling
  - Load management
  - Caching



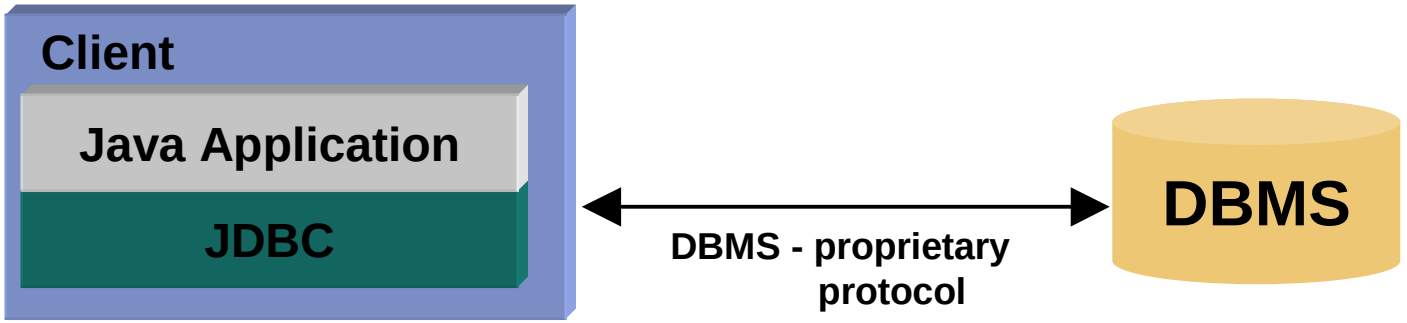
Type 4 Drivers

- Type 4 drivers are ‘all-Java’ driver implementations that *do not* require client side configuration.



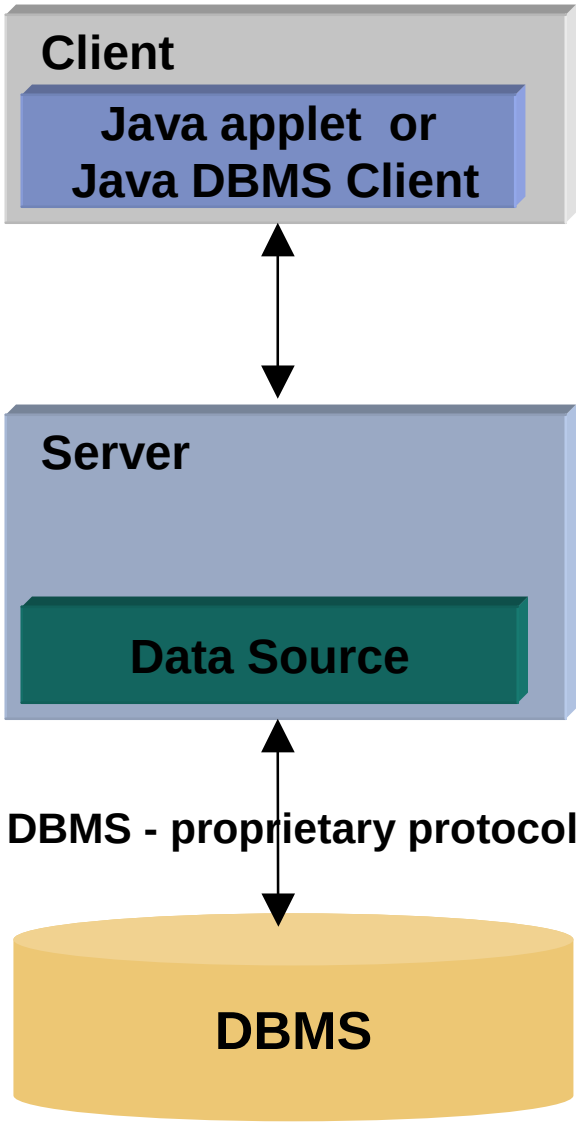
Two-Tier Architecture

- In the two-tier model a Java application communicates directly with the DBMS.
- A JDBC driver is needed that can communicate directly with the DBMS.
- This is a client/server configuration.



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.



WebLogic Server Two-Tier Drivers

- The following table shows the two-tier drivers for vendor specific platforms that WebLogic Server provides.

Driver	Type
DB2	4
Informix	4
SQL Server	4
Oracle	4
Sybase	4
MySQL	4

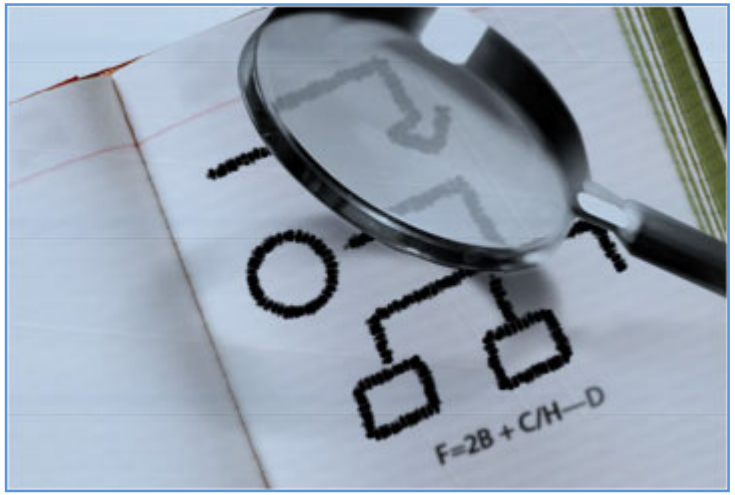
Choosing the Correct Driver

- Choosing the correct driver can have significant impact on performance.
- For two-tier applications use the type 1, 2 or 4 driver specific to the DBMS you are using.
- For multi-tier applications use:
  - Data Source lookup in a client class
  - A type 1, 2 or 4 driver on the server, specific to the DBMS you are using
  - XA driver where transaction support is required

Section Review

In this section we discussed:

- ✓ JDBC high level architecture
- ✓ WebLogic Server driver types and support
- ✓ Two-tier versus multi-tier architectures
- ✓ Selecting a driver type



Road Map

1. Overview of JDBC
2. **Data Sources**
   - Describe a Data Source and How It Works
   - Use the Administration Console to Create a Data Source
3. Monitoring and Logging

What Is a Data Source?

- A Data Source object provides a way for a JDBC client to obtain a database connection from a connection pool.
- A Data Source:
  - Is stored in the WLS JNDI tree
  - Can support transactions
  - Is associated with a connection pool

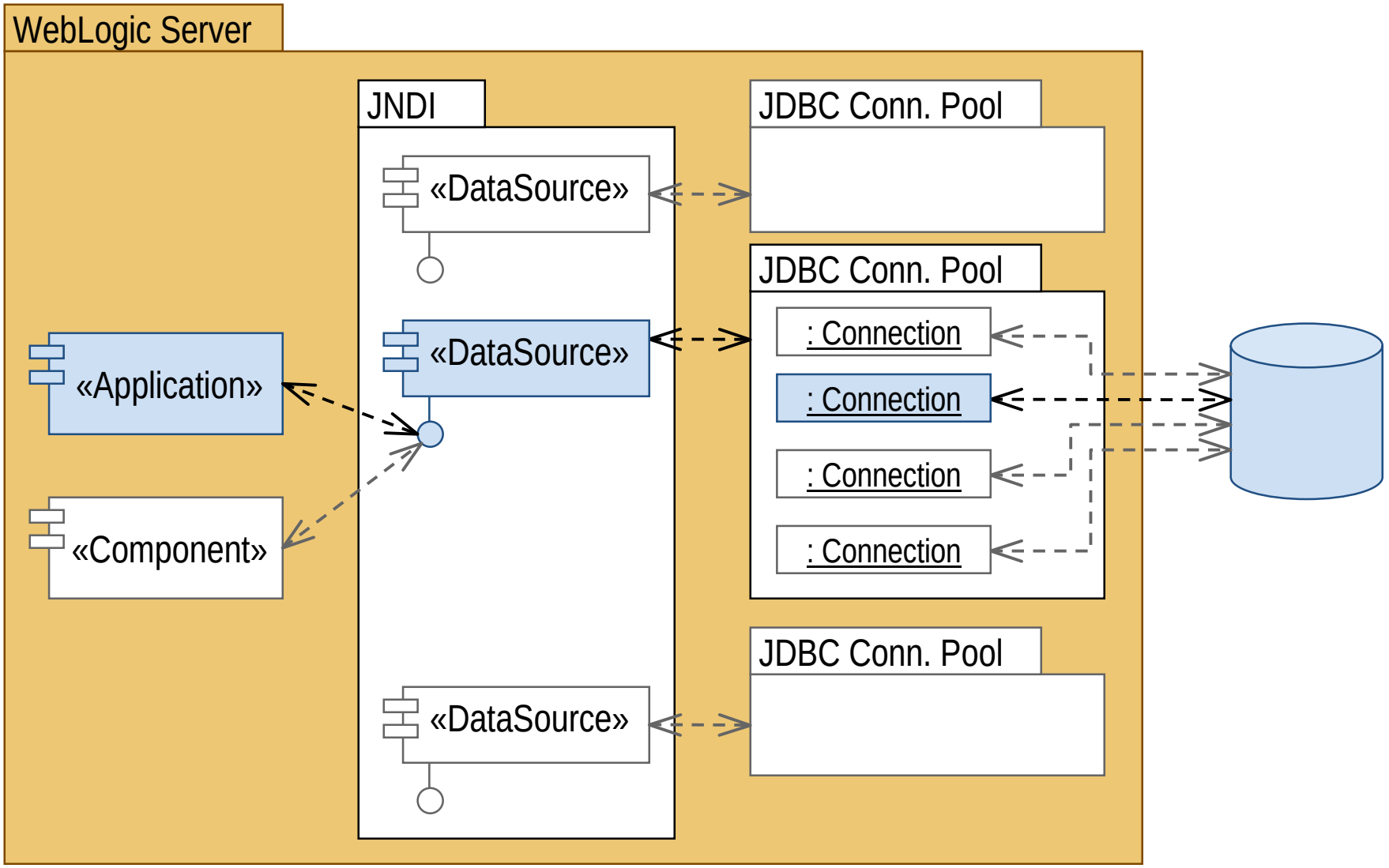
What Is a Connection Pool?

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

Benefits of Data Sources + Connection Pools

- Some advantages of this approach are:
  - Time and overhead are saved by using an existing database connection
  - Connection information is managed in one location in the Administration Console
  - The number of connections to a database can be controlled
  - The DBMS can be changed without the application developer having to modify underlying code
- A connection pool allows an application to “borrow” a DBMS connection.

JDBC Data Source Architecture

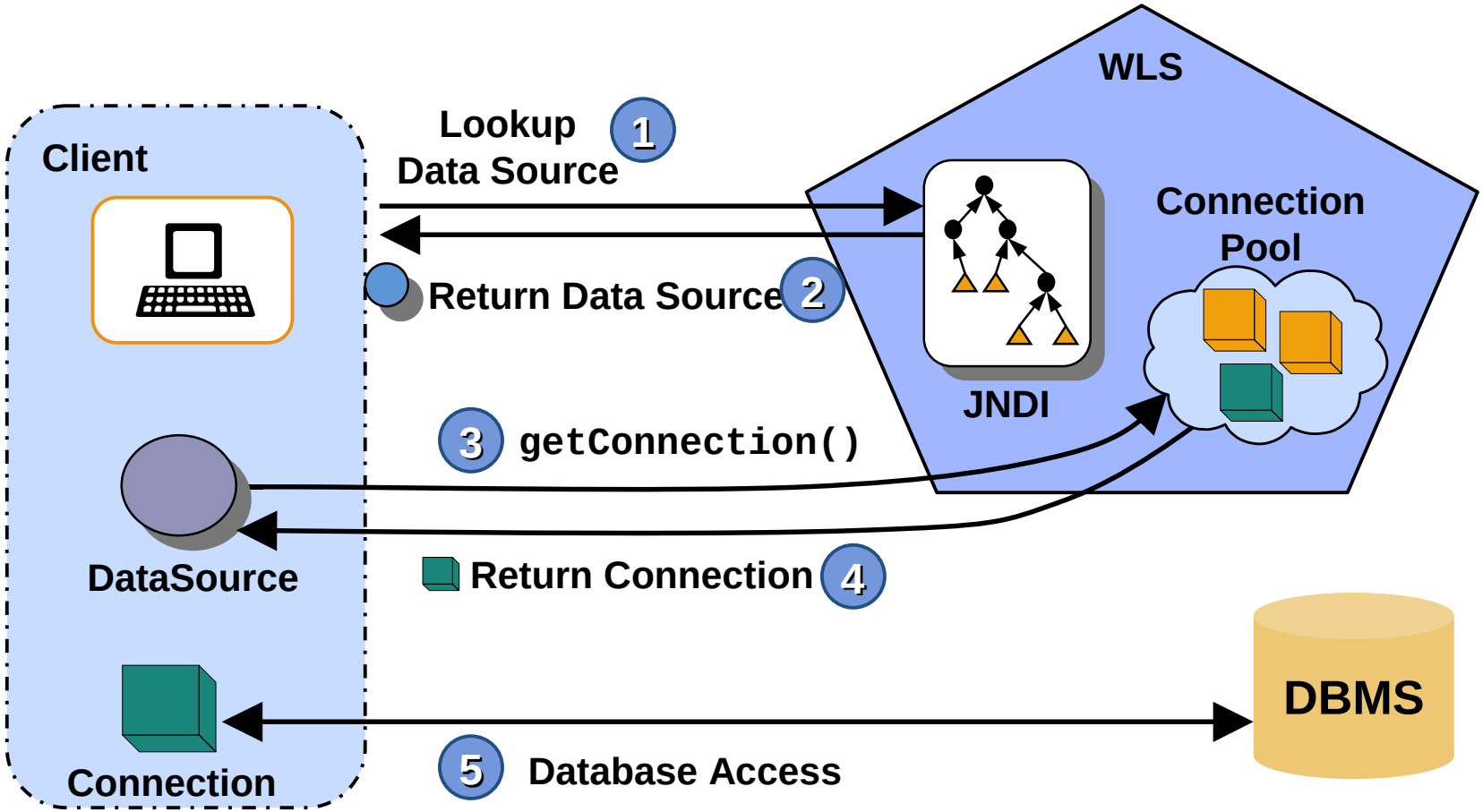


Modular Configuration and Deployment of JDBC Resources

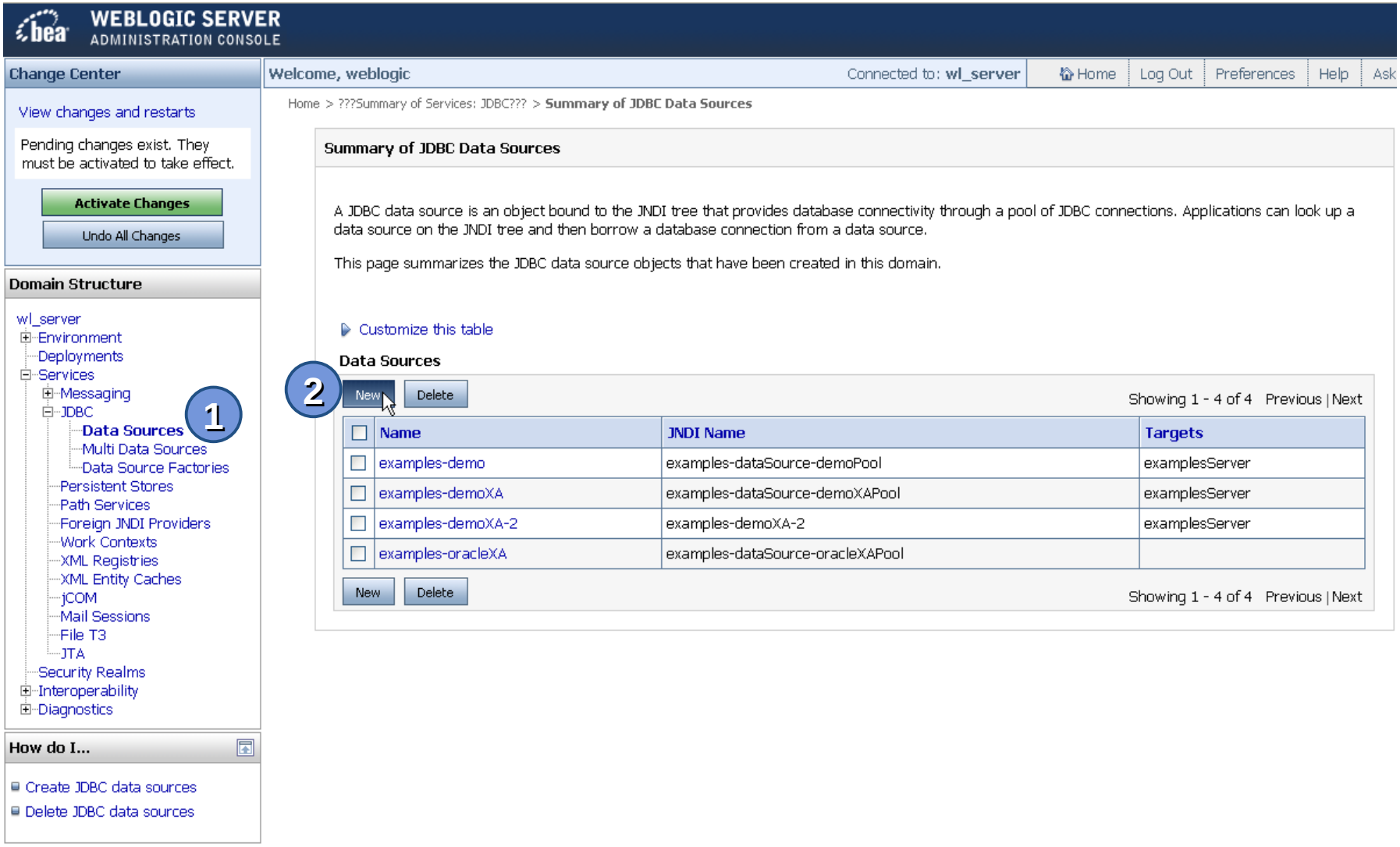
- JDBC configurations in WLS are stored in XML documents:
  - All JDBC configurations must conform to the new `weblogic-jdbc.xsd` schema.
  - IDEs and other tools can validate JDBC modules based on the schema.
- You create and manage JDBC resources either as system modules or as application modules.
- JDBC *application modules* are a WLS-specific extension of J2EE modules and can be deployed either within a J2EE application or as stand-alone modules.

How Data Sources Are Used

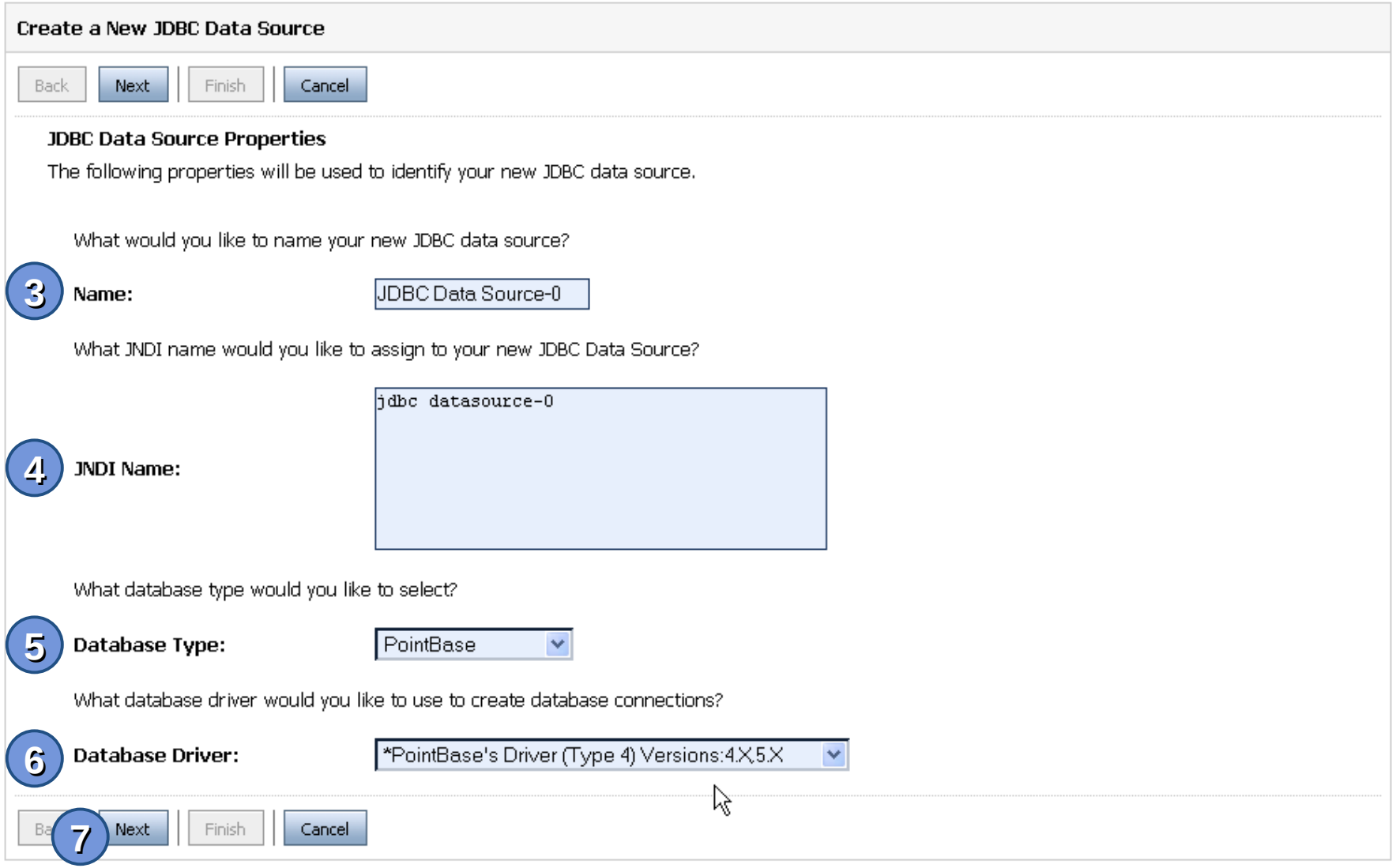
- A client retrieves a Data Source through JNDI look up and uses it to obtain a database connection.



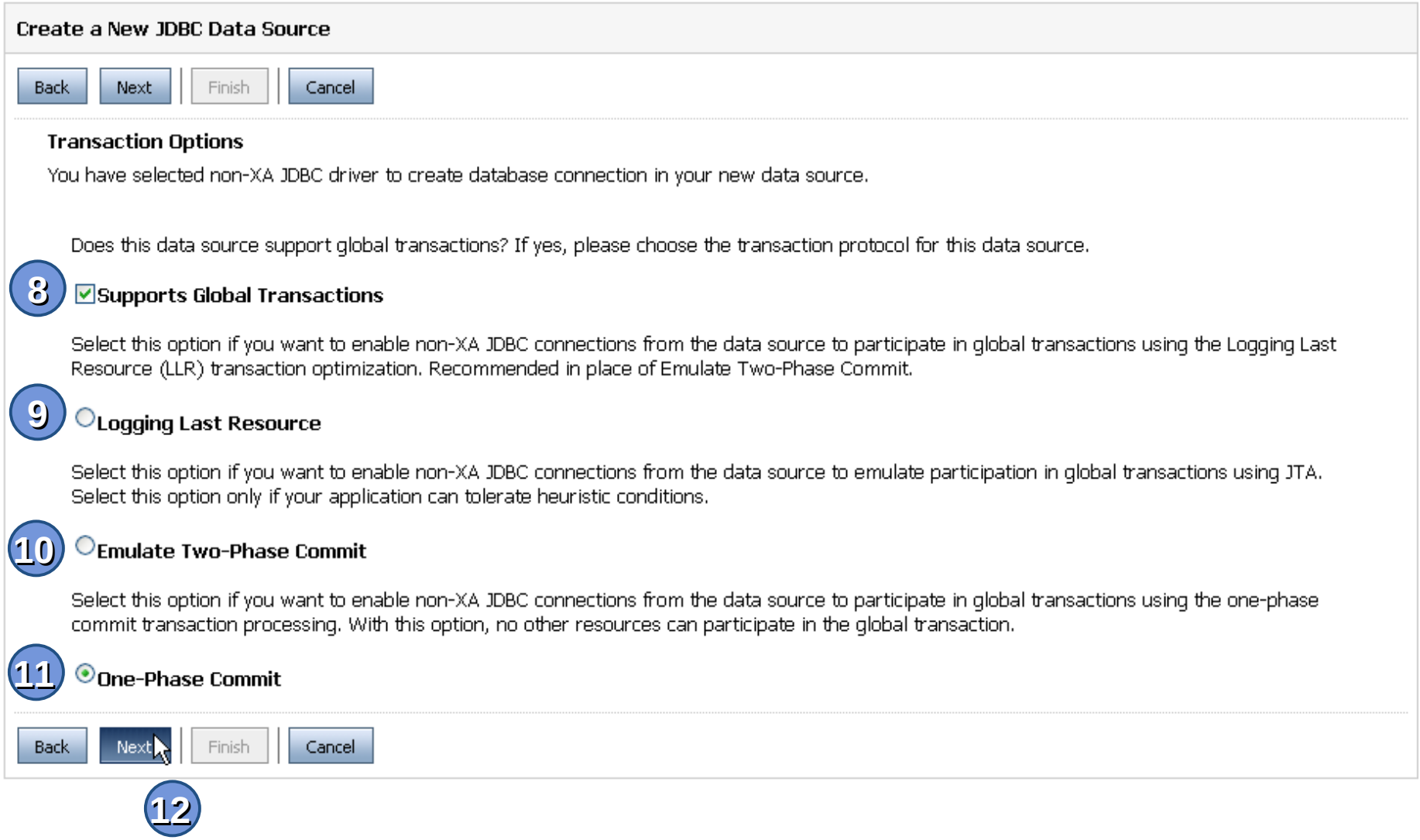
Creating a JDBC Data Source…



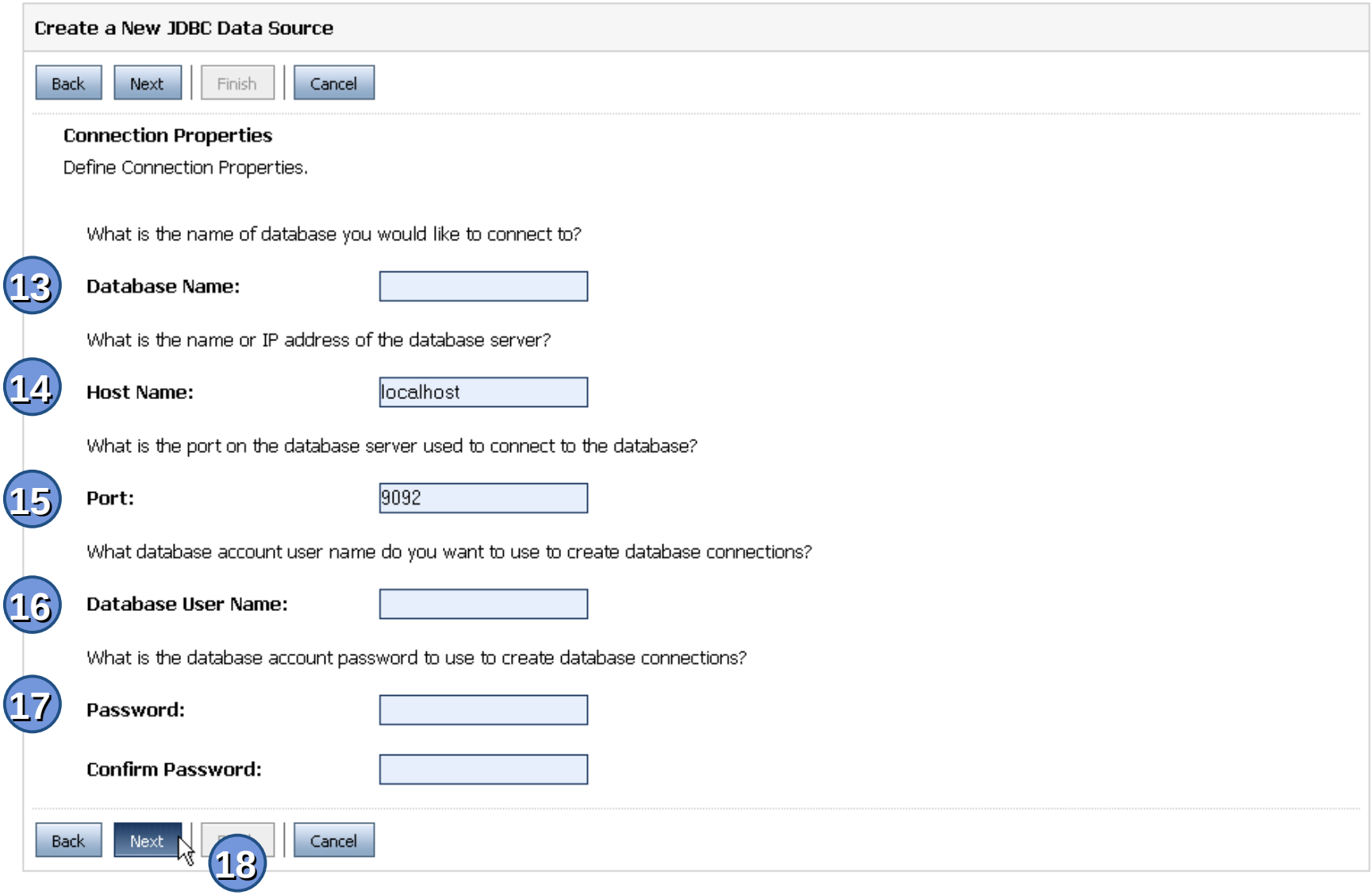
Creating a Data Source: Properties



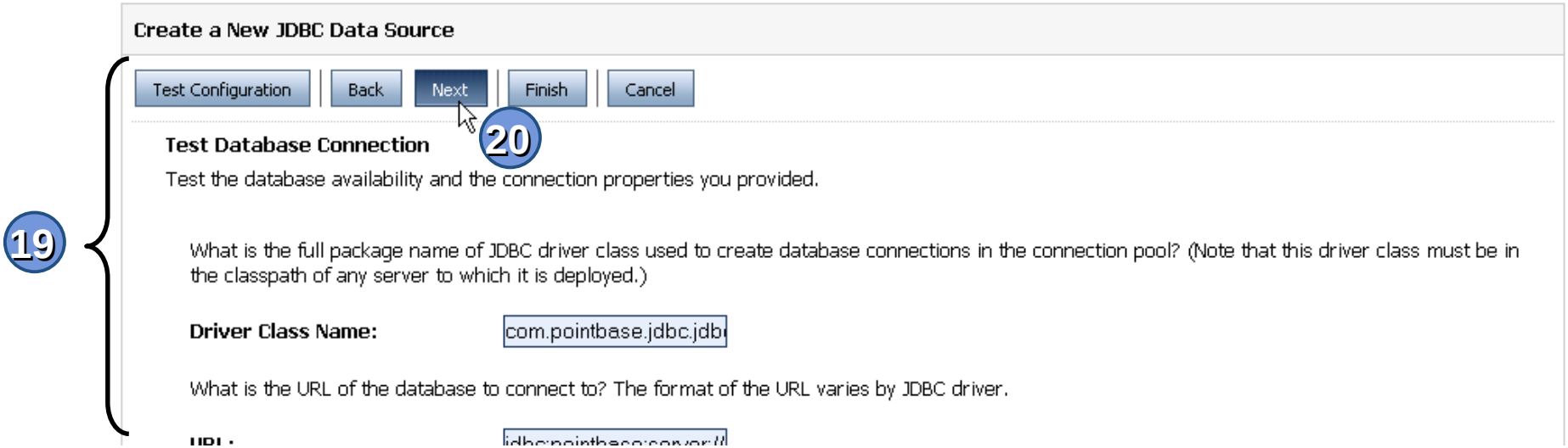
Creating a Data Source: XA Options

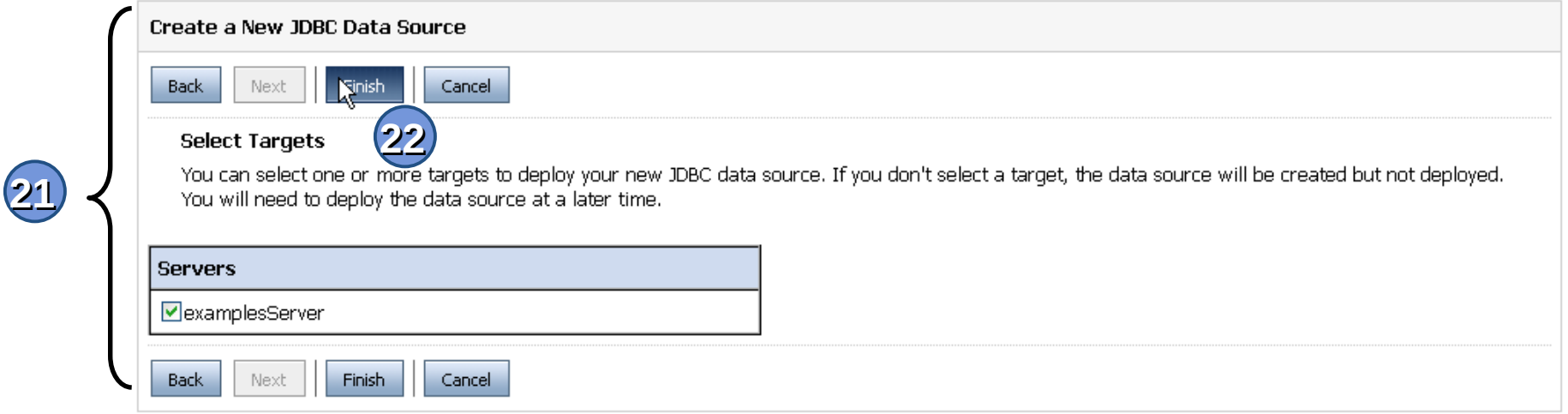


Creating a Data Source: Connection Properties

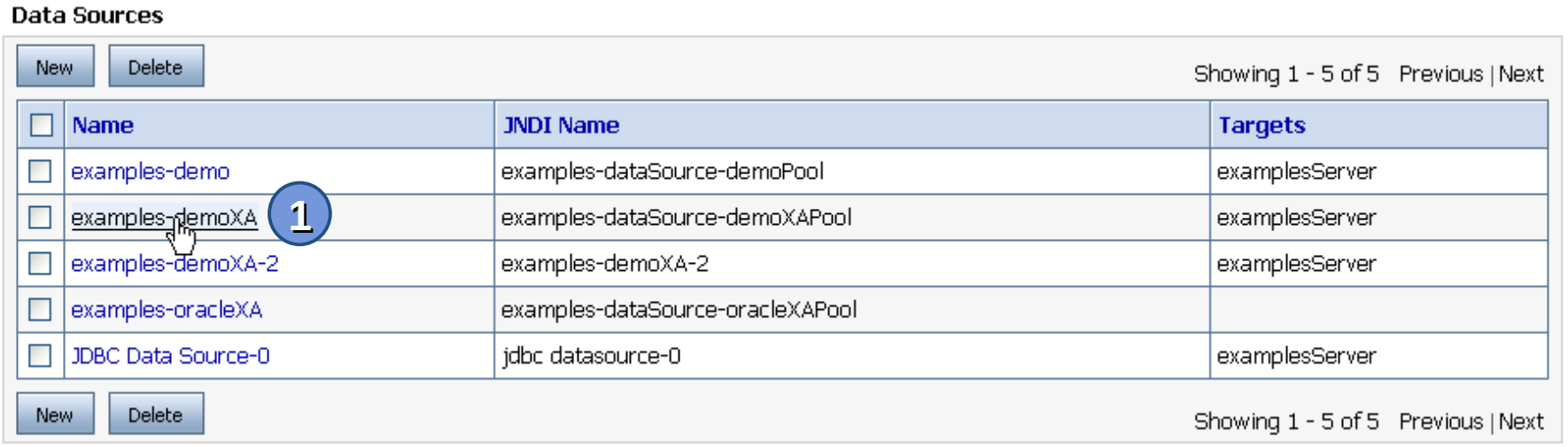


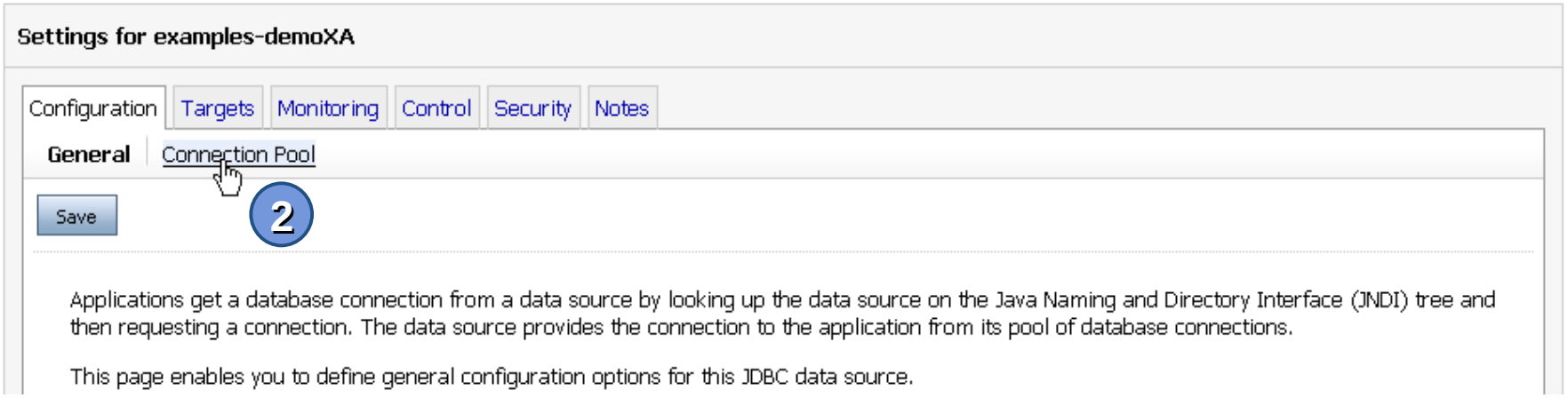
Creating a Data Source: Test Database Connections & Select Targets



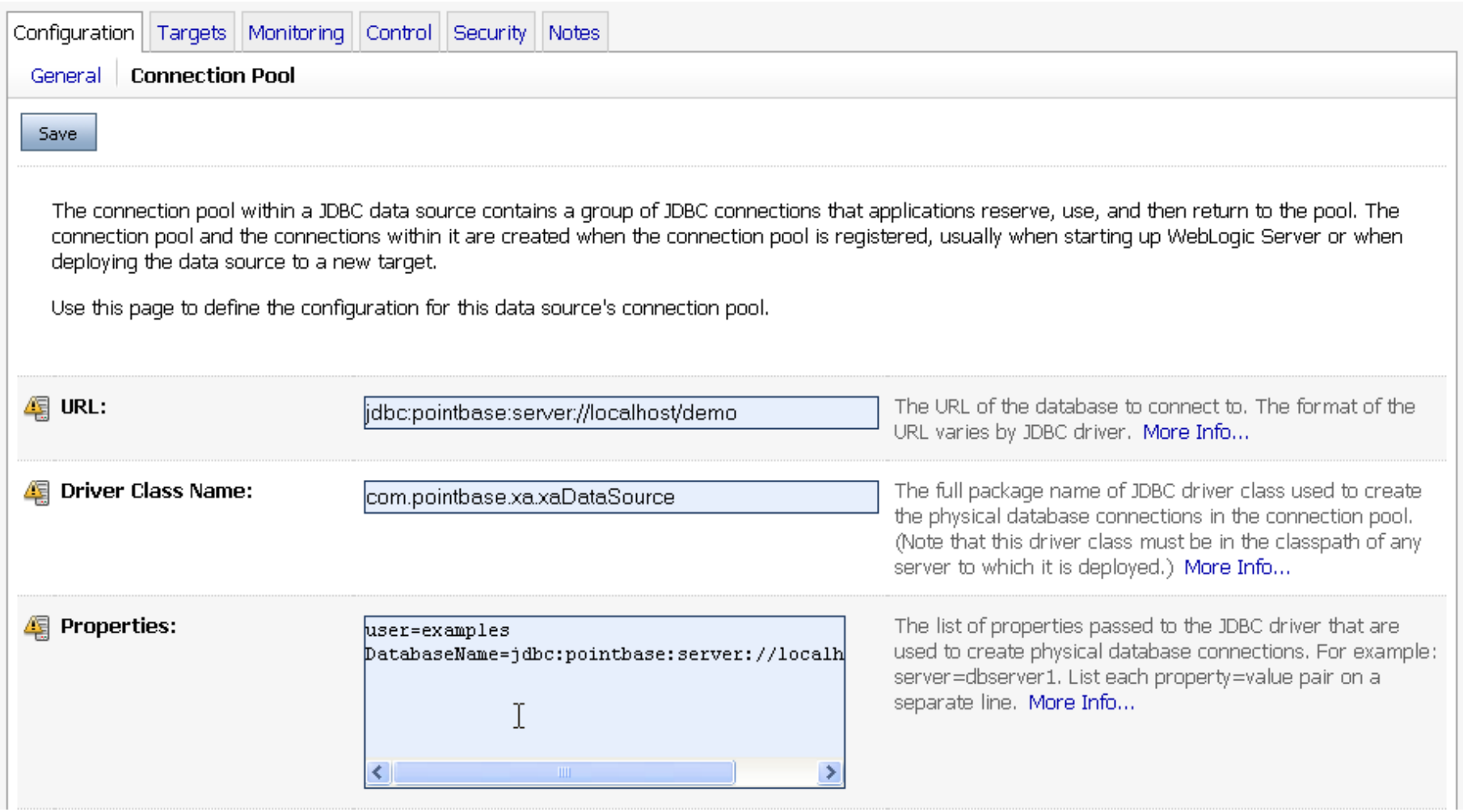


Configuring a Connection Pool…

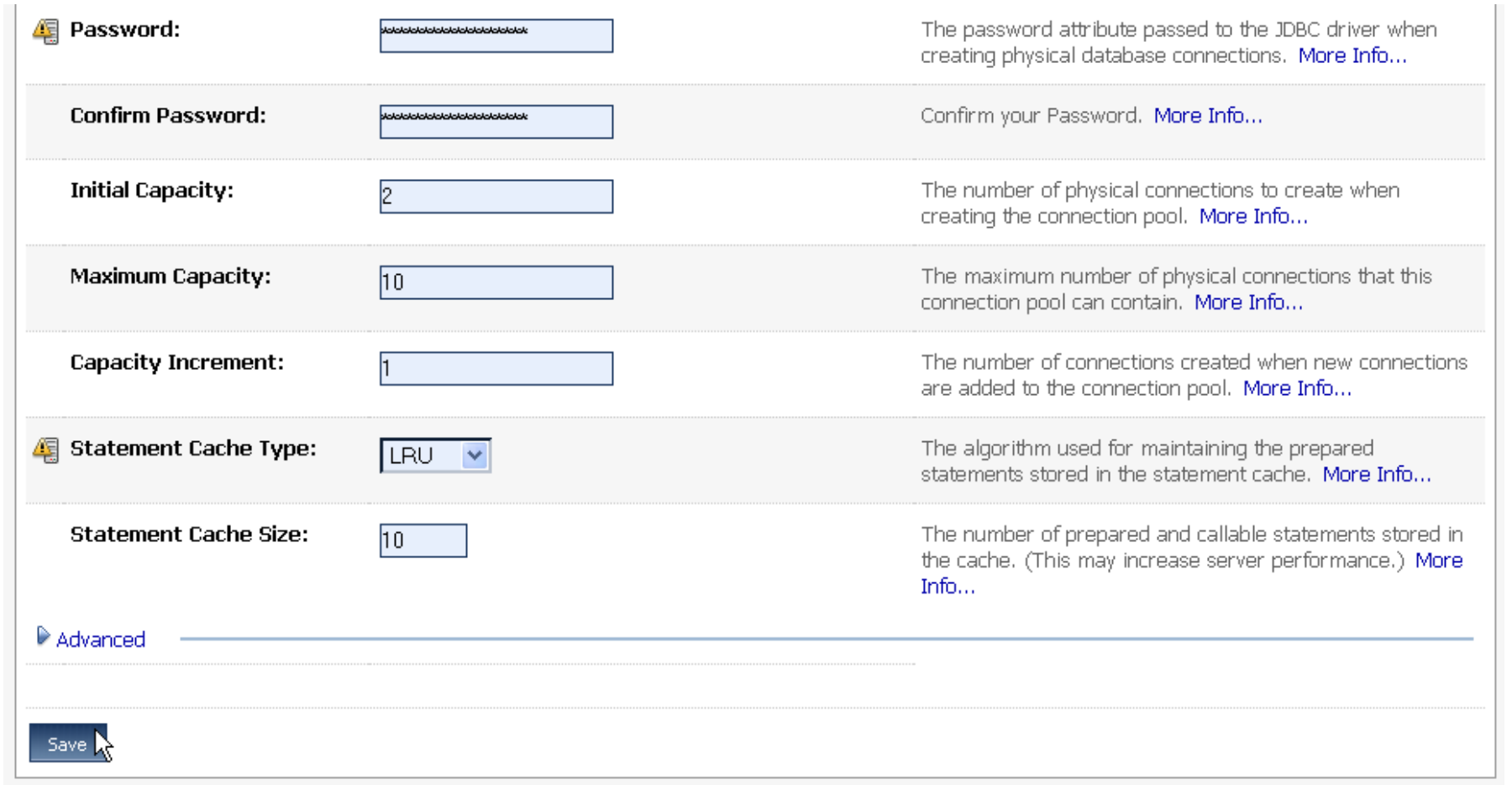




Configuring Settings of a Connection Pool…



...Configuring Settings of a Connection Pool



Connection Pool Checklist

- You can modify a connection pool after the Data Source has been created
- Before modifying a connection pool, you should understand:
  - The JDBC URL of the database
  - The connection properties used to authenticate a user or optionally configure the driver
- Ask your DBA for:
  - The maximum number of connections your application will be allowed

JDBC URLs

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

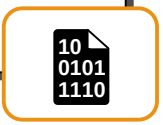
The syntax for a JDBC URL is:

```
jdbc:<subprotocol>:<subname>
```

Where:

<subprotocol> identifies the database connectivity mechanism

<subname> identifies the data source. The subname can vary, depending on the subprotocol



JDBC URL Examples

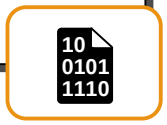
Example 1. This URL could be used to access a Pointbase database:

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

The subprotocol is “pointbase:server”, the subname is a location of Pointbase database named “HRDATABASE”.

Example 2. This URL specifies that the “oracle:thin” subprotocol should be used to connect to an Oracle database:

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



Connection Properties…

- Connection properties are:
  - Key/value pairs
  - Used to configure JDBC connections and are passed to the driver during connection setup

For a complete list see your driver documentation!

…Connection Properties

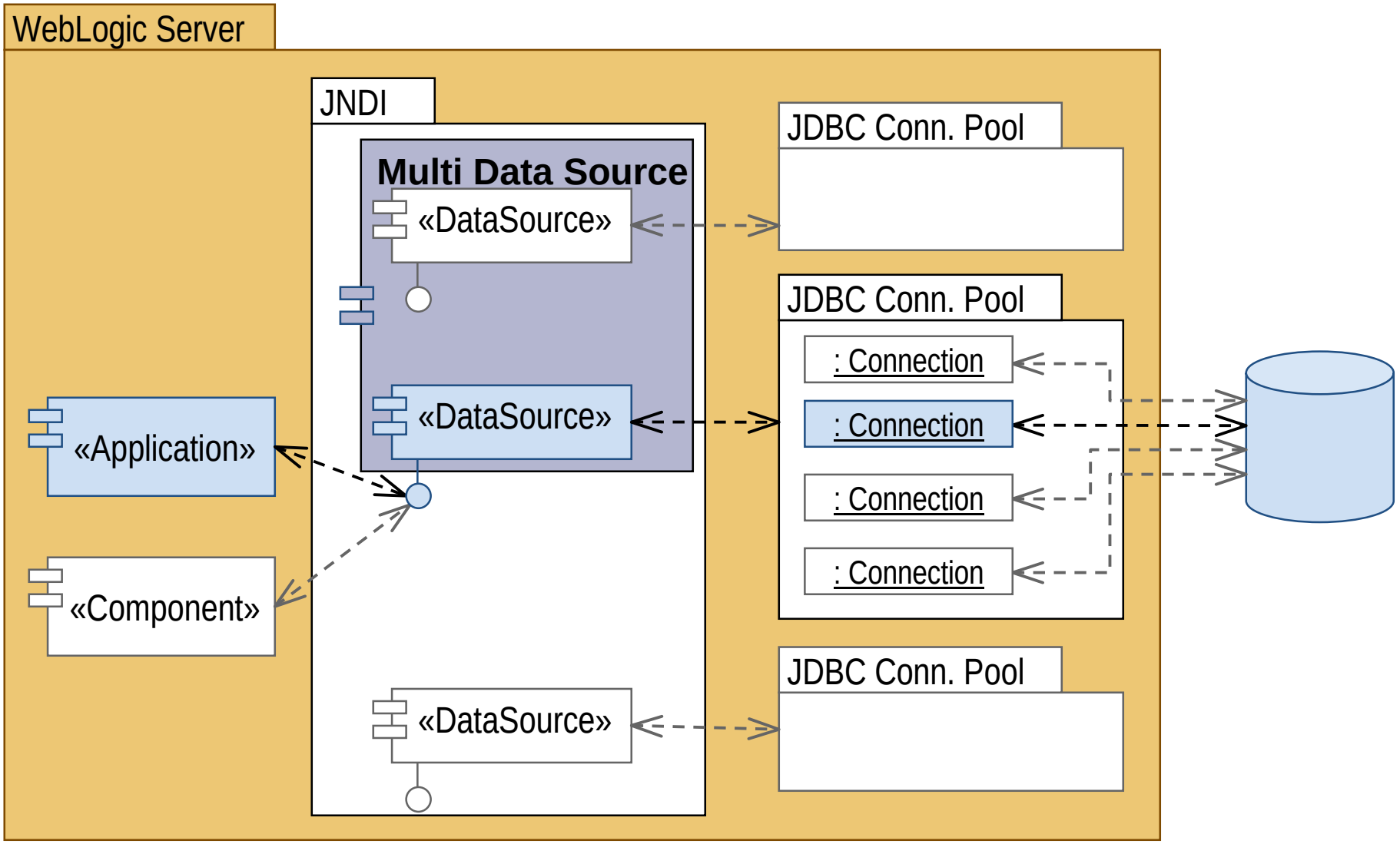
- A partial list of connection properties for the supplied drivers:

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`

Multi Data Sources…

- Multi Data Source:
  - Is an abstraction around a group of data sources
  - Determines which data source to use to satisfy the request depending on the algorithm selected in the multi data source configuration:
    - Load balancing
      Or
    - Failover
  - Are bound to the JNDI tree
- XA Support for Multi Data Sources
  - The WLS JDBC supports using Multi Data Sources in XA transactions.
  - You can configure the data sources contained within the Multi Data Source to use XA JDBC drivers.

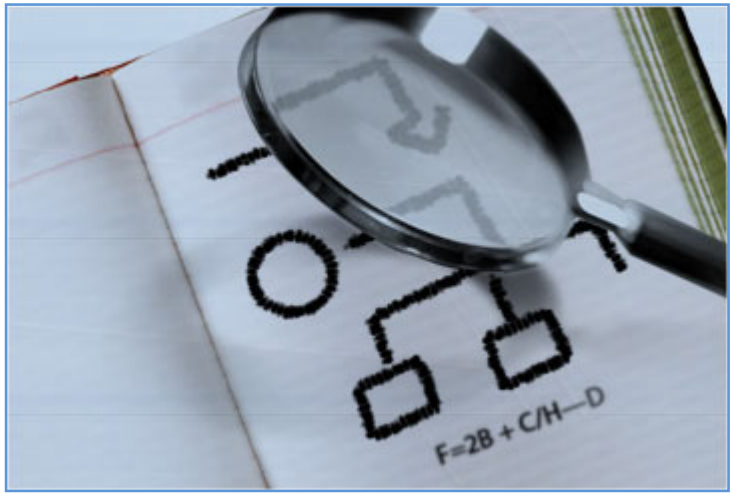
… Multi Data Sources



Section Review

In this section we discussed:

- ✓ Data Source definition and workings
- ✓ Data Source management with the Administration Console
- ✓ Multi Data Sources

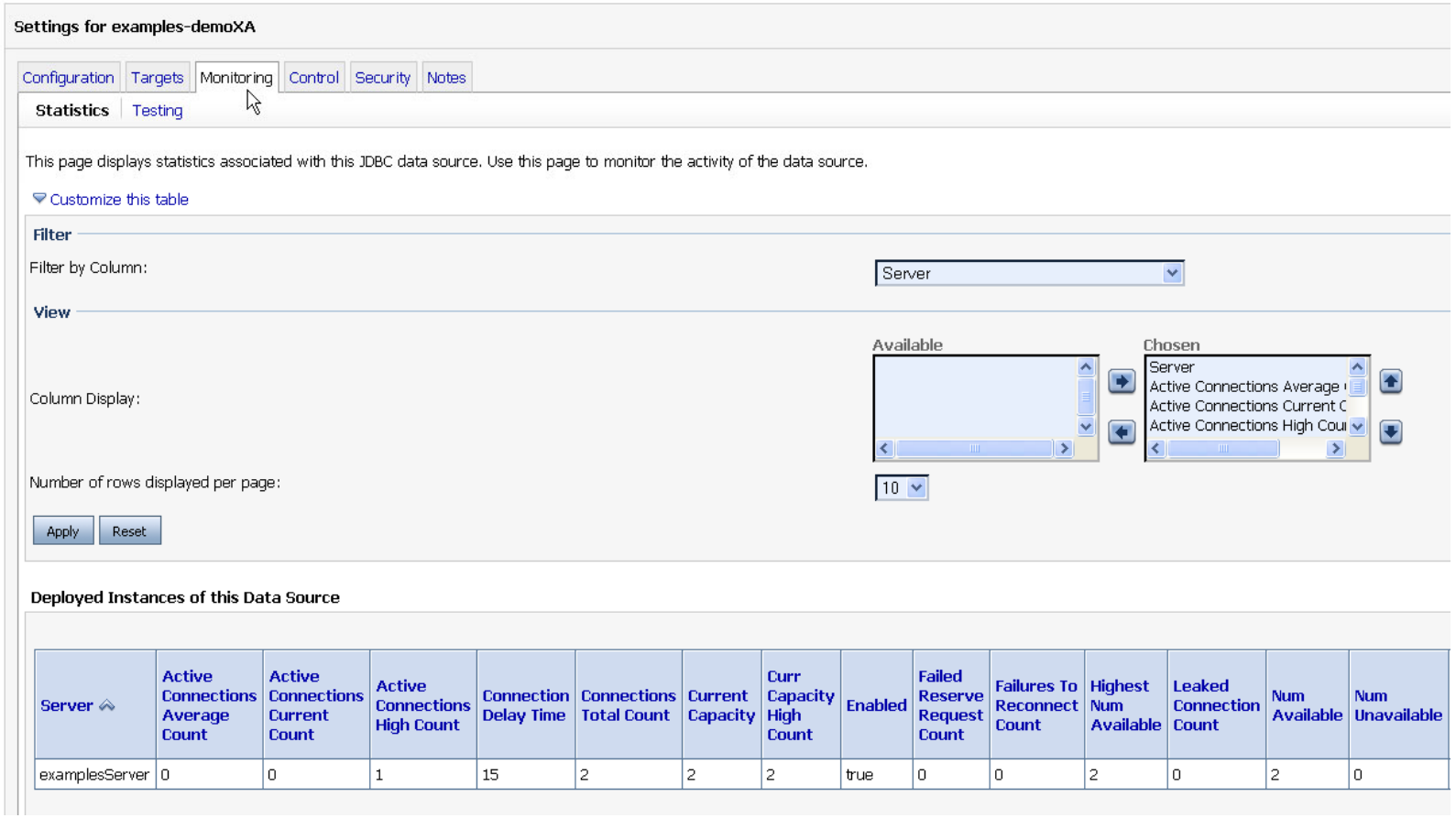


Road Map

1. Overview of JDBC
2. Data Sources
3. **Monitoring & Testing Data Sources**
   - Monitoring
   - Testing

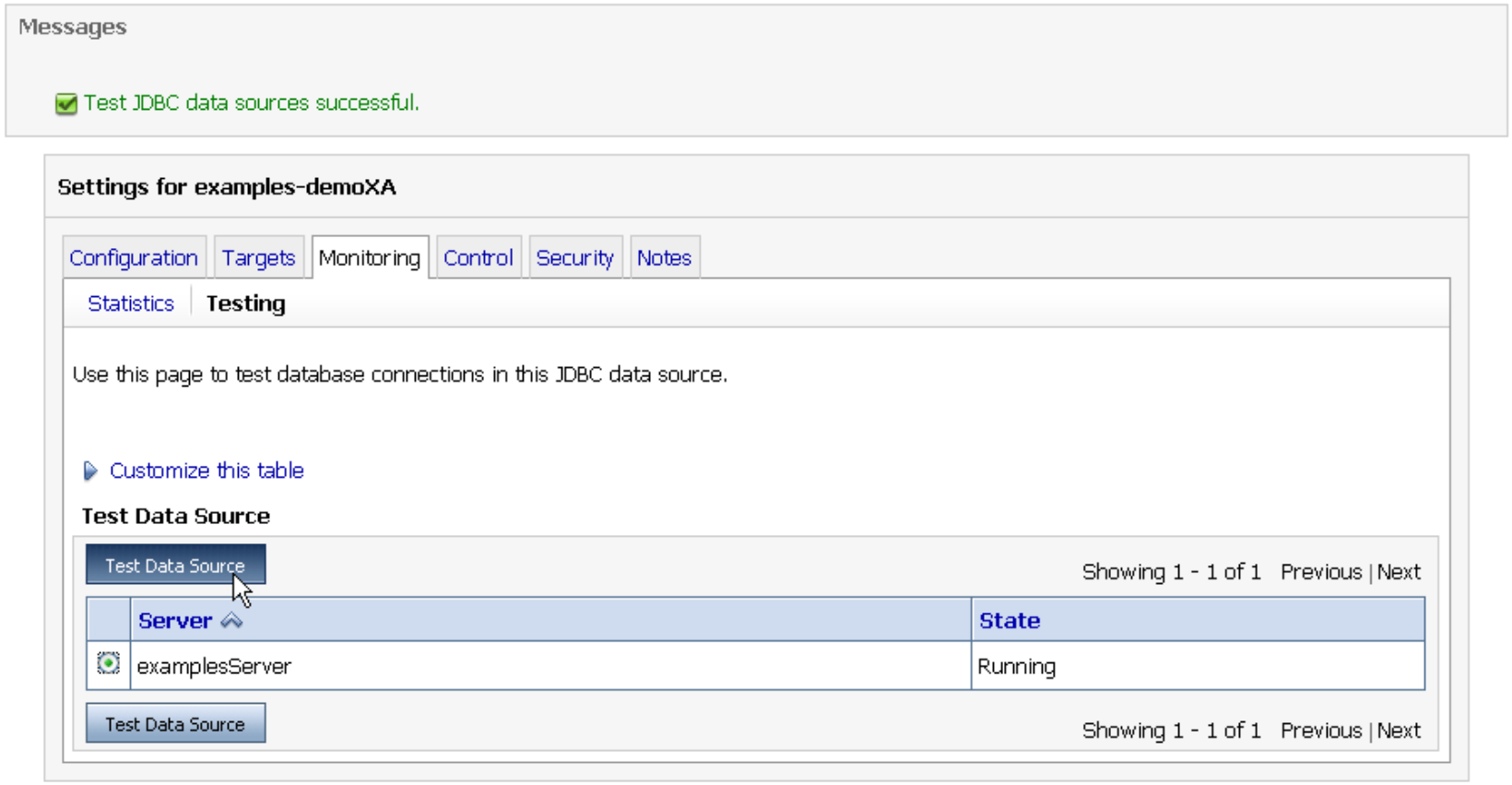
Monitoring Data Sources: Statistics

- The Administration console provides two types of data source monitoring: statistics and testing.



Monitoring Data Sources: Testing

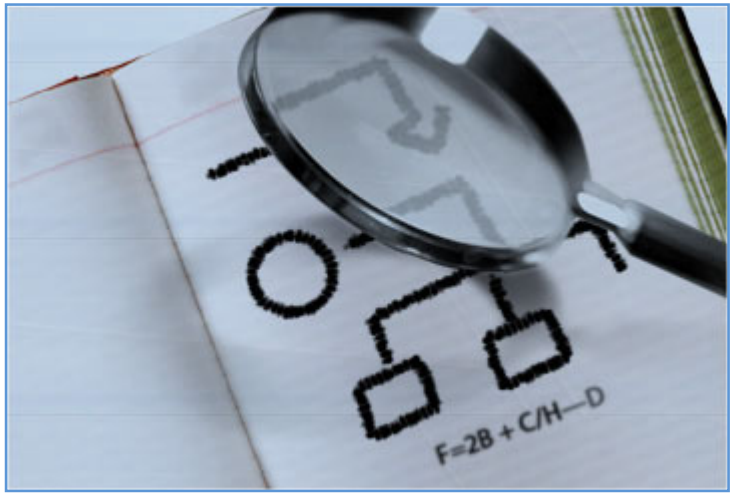
- The Administration console provides a mechanism for manually testing individual data sources.



Section Review

In this section we discussed:

- ✓ Monitoring statistics and the testing of an individual data source.



Exercise

Configuring Data Sources

- In this lab you will work with configuring and monitoring Data Sources.
- Ask the instructor for any clarification.
- The instructor will determine the stop time.

Lab Exercise

Module Review

In this module we discussed:

- ✓ JDBC high level architecture
- ✓ WebLogic Server provided JDBC driver types
- ✓ Data Source definition and workings
- ✓ Connection pool definition and workings
- ✓ Managing JDBC resources with the Administration Console