

Configuring Oracle TimesTen In-Memory Database 11.2.2 for J2EE Application Servers and Object-Relational Mapping Frameworks

*An Oracle White Paper
April 2014*

Configuring Oracle TimesTen In-Memory Database 11.2.2 for J2EE Application Servers and Object- Relational Mapping Frameworks

Introduction	3
Conventions	3
Prerequisites	3
Troubleshooting	4
TimesTen for JBoss EAP 6.2	6
Configuring the TimesTen JDBC driver	6
Configuring TimesTen data sources	8
Using JBoss JPA with TimesTen	12
TimesTen for Oracle WebLogic Server 12.1	13
Configuring the TimesTen JDBC driver	13
Configuring TimesTen data sources	13
Using WebLogic Server JPA with TimesTen	19
TimesTen for Oracle GlassFish Server 3.1	20
Configuring the TimesTen JDBC driver	20
Configuring TimesTen connections	20
Configuring a TimesTen JDBC connection pool	21
Configuring a TimesTen JDBC resource	23
Using Oracle GlassFish Server JPA with TimesTen	24
TimesTen for WebSphere Application Server 8.5	25
Configuring TimesTen connections	25
Configuring a TimesTen JDBC provider	26
Configuring TimesTen data sources	28
Using WebSphere Application Server JPA with TimesTen	30
TimesTen for EclipseLink 2	31
Configuring the TimesTen JDBC driver for EclipseLink	31
TimesTen for Hibernate 4	32
Configuring the TimesTen JDBC driver for Hibernate	32
Hibernate configuration properties for TimesTen	32
TimesTen for OpenJPA 2	35
Configuring the TimesTen JDBC driver for OpenJPA	35

Configuring Oracle TimesTen In-Memory Database 11.2.2 for J2EE Application Servers and Object- Relational Mapping Frameworks

INTRODUCTION

This white paper is for application developers who use and administer TimesTen JDBC and for system administrators who configure and manage the TimesTen database. It provides information about configuring J2EE application servers and object-relational mapping frameworks for use with TimesTen 11.2.2.

Additional information and code examples demonstrating the use of TimesTen with Java technologies are available in the TimesTen Quick Start. The TimesTen Quick Start is an optional component of a TimesTen installation.

CONVENTIONS

This document uses the following conventions:

- *tt_install_dir*: The path to the directory where TimesTen is installed.
- *gs_install_dir*: The path to the directory where Oracle GlassFish Server is installed.
- *gs_domain*: The name of the directory that contains an Oracle GlassFish Server domain configuration.
- *jboss_install_dir*: The path to the directory where JBoss Enterprise Application Platform is installed.
- *wl_domain_dir*: The path to the directory that contains a WebLogic Server domain configuration.

PREREQUISITES

The sample configurations provided in this document require that you install both TimesTen and the target J2EE application server or object-relational mapping framework on the same machine. In addition, the following TimesTen items are required:

- A TimesTen direct driver DSN called `SAMPLEDB_1122`. This is already defined in the `.odbc.ini` ODBC configuration file or in the ODBC Data Source Administrator dialog on Windows platforms.
- A TimesTen client/server driver DSN called `SAMPLEDBCS_1122`. This is already defined in the `.odbc.ini` ODBC configuration file or in the ODBC Data Source Administrator dialog on Windows platforms. The `SAMPLEDBCS_1122` DSN should be configured to connect to the database defined by the `SAMPLEDB_1122` direct driver DSN.
- An internal TimesTen user account with the user name 'scott' and password 'tiger' needs to be defined in the database associated with the `SAMPLEDB_1122` DSN.

Test and configure the two DSNs and the `scott/tiger` user account with the TimesTen `ttIsql` utility before following the procedures and using the configurations included in this document.

TROUBLESHOOTING

- The TimesTen JDBC driver loads native shared libraries at runtime. This requires the Java Virtual Machine (JVM) to use the same architecture as the TimesTen JDBC driver. For example, a 32-bit JVM cannot connect to a TimesTen database using a 64-bit version of the TimesTen JDBC driver and vice versa. Attempts to mix architectures in this way results in a runtime error when the Java application attempts to connect to the TimesTen database.
- The TimesTen JDBC driver includes a tracing facility that can help diagnose problems. To enable the tracing facility you must enable a Java system property called `timesten.tracefile` for the JVM accessing the TimesTen JDBC driver. You can enable tracing from the Java command line with the syntax `-Dtimesten.tracefile=filename` where *filename* is the complete path to a local output file.
- Attempts to establish a connection to a TimesTen database from within the JVM may result in a TimesTen JDBC driver exception with the message: “Cannot attach database shared memory segment”. This problem is likely to occur on 32-bit platforms when the system is unable to allocate or map enough contiguous shared memory to hold the TimesTen database. It can also occur when there is a conflict with the value of the `PLSQL_MEMORY_ADDRESS` TimesTen connection attribute. For detailed instructions on how to resolve this and other similar connection problems see the *Oracle TimesTen In-Memory Database Troubleshooting Procedures Guide* and the *Oracle TimesTen In-Memory Database Installation Guide*. These are the recommended troubleshooting steps:
 - a. Verify that it is possible to connect to the TimesTen database outside of the JVM. You can use the `ttIsql` utility for this purpose.

- b. Try increasing the maximum amount of shared memory available to the system. See the *Oracle TimesTen In-Memory Database Installation Guide* for platform specific details.
- c. Try decreasing the memory size of the TimesTen database by recreating the database with lower values for the `PermSize` and `TempSize` connection attributes.
- d. Try adjusting the value of the `PLSQL_MEMORY_ADDRESS` connection attribute. If PL/SQL support is not required for the TimesTen database, then disable it by recreating the TimesTen database with the `PLSQL` connection attribute set to 0.
- e. Terminate any unnecessary processes on the system to make more memory available.
- f. If possible, try a client/server TimesTen JDBC connection instead of a direct connection.
- g. If possible, try a 64-bit version of the TimesTen JDBC driver in conjunction with a 64-bit JVM.

TIMESTEN FOR JBOSS EAP 6.2

This section provides descriptions and examples of configuring the TimesTen JDBC driver for use with JBoss Enterprise Application Platform 6.2. This section requires that you install both TimesTen and JBoss EAP on the same machine, and that a standalone JBoss server instance is in use.

Configuring the TimesTen JDBC driver

Before you can use JBoss with TimesTen databases, the JBoss server environment must have access to the TimesTen shared libraries and the TimesTen JDBC driver jar file.

1. Set the `LD_LIBRARY_PATH` environment variable (or the equivalent variable for your OS) to include the `tt_install_dir/lib` directory where TimesTen shared libraries are located. On Windows, set the `PATH` environment variable to the `tt_install_dir/bin` directory where the TimesTen DLLs are located. This variable must be set for the environment where JBoss EAP runs.
2. Restart the JBoss server for the environment changes to take effect.
3. Log in to the JBoss management console. The default web address is `http://localhost:9990/console`.
4. In the top menu of the JBoss management console, select **Runtime**.
5. In the top-left pane of the Runtime tab, expand the **Server** node and, then select **Manage Deployments**.
6. In the DEPLOYMENTS tab, click **Add**, and then click **Choose File** in the Create Deployment dialog.
7. Specify the location of the TimesTen JDBC driver jar file. If JBoss server is using JDK 1.7, then select the Java 7 runtime version of the TimesTen JDBC driver jar file located at `tt_install_dir/lib/ttjdbc7.jar`. See Figure 1.1. If JBoss server is using JDK 1.6, then select the `tt_install_dir/lib/ttjdbc6.jar` file.

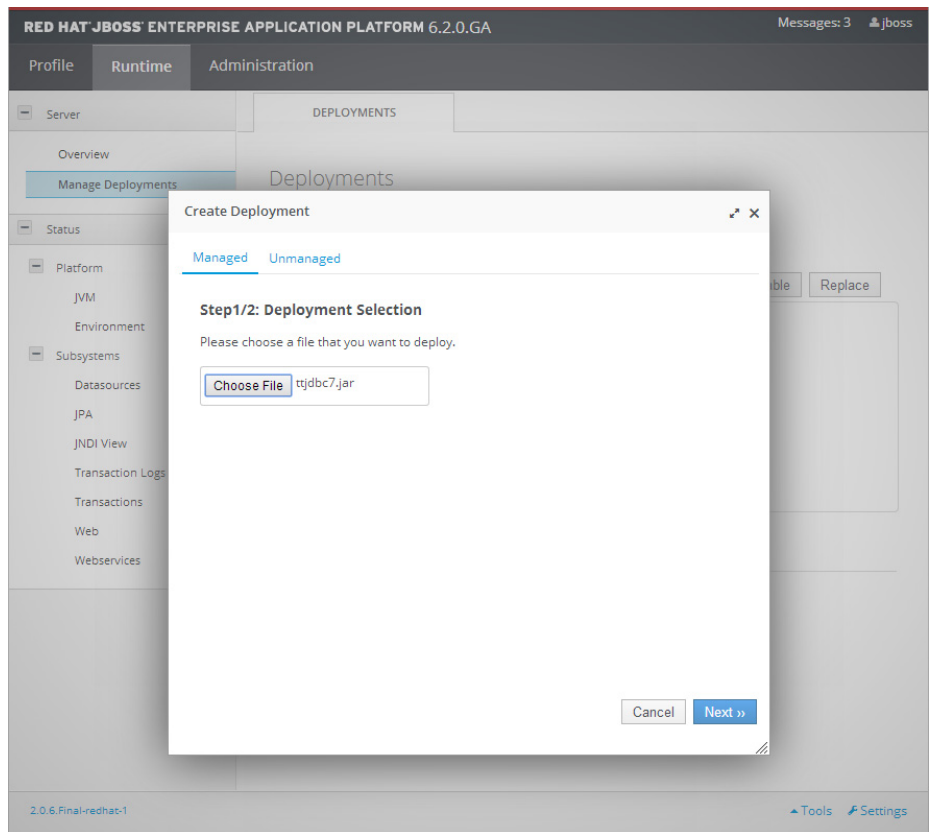


Figure 1.1 Deployment Selection dialog

8. Click **Next**.
9. Click **Save**.
10. In the DEPLOYMENTS tab, a new deployment called `ttjdbc7.jar` or `ttjdbc6.jar` is now displayed. Select **En/Disable**. See Figure 1.2. Click **Confirm** to enable the TimesTen JDBC driver.

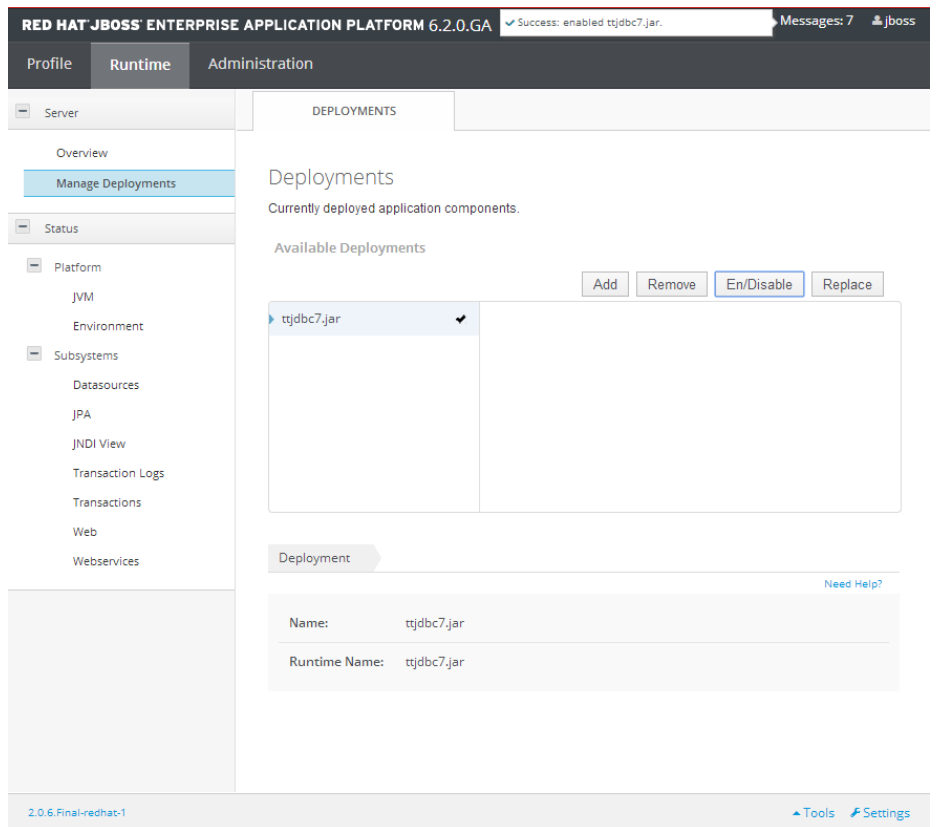


Figure 1.2 Available Deployments

Configuring TimesTen data sources

The TimesTen JDBC driver supports four different types of connections to TimesTen databases.

- **Direct access with local transactions only** - This configuration provides the fastest database performance. The TimesTen database must reside on the same machine as the JBoss server.
- **Direct access with XA distributed transaction support** - This configuration uses JTA to support distributed transactions. The TimesTen database must reside on the same machine as the JBoss server.
- **Client/server access with local transactions only** - The TimesTen database can reside on a remote machine that is running the TimesTen server.
- **Client/server access with XA distributed transaction support** - This configuration uses JTA to support distributed transactions. The TimesTen database can reside on a remote machine that is running the TimesTen server.

Each type of TimesTen JDBC driver connection requires a unique JBoss data source configuration. The key difference between the configurations is the TimesTen driver class name and the TimesTen URL. TimesTen connections that support local

transactions use the `com.timesten.jdbc.TimesTenDriver` class. Connections that require distributed transaction support through JTA use the `com.timesten.jdbc.xa.TimesTenXADataSource` class.

The next procedure describes how to configure a direct access TimesTen data source that supports local transactions by the use of the JBoss management console.

1. Log in to the JBoss management console. The default web address is `http://localhost:9990/console`.
2. In the top menu of the JBoss management console, select **Profile**.
3. In the top-left pane of the Profile tab, expand the **Subsystems** node, then expand **Connector** node, and then select **Datasources**.
4. Click **Add**.
5. In the Datasource Attributes step of the Create Datasource dialog, enter a name and a JNDI name for the new data source. See Figure 1.3.

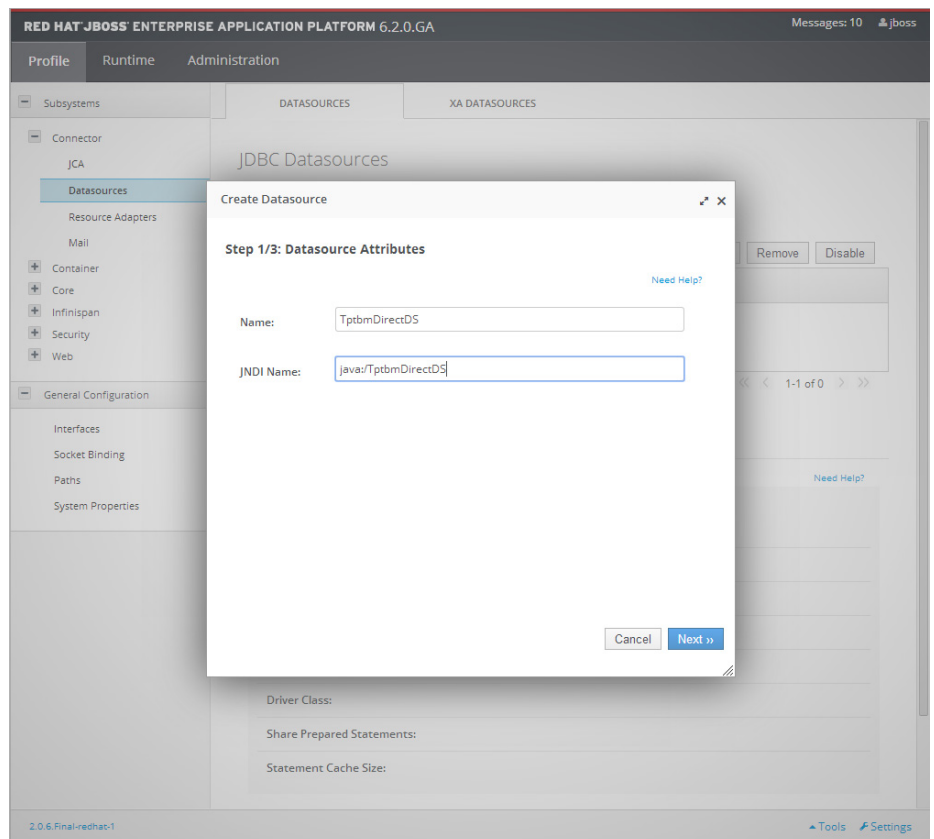


Figure 1.3 Step 1/3: Datasource Attributes

6. Click **Next**.

7. In the JDBC Driver step of the Create Datasource dialog, select the TimesTen JDBC driver deployment called `ttjdbc7.jar` or `ttjdbc6.jar`. See Figure 1.4.

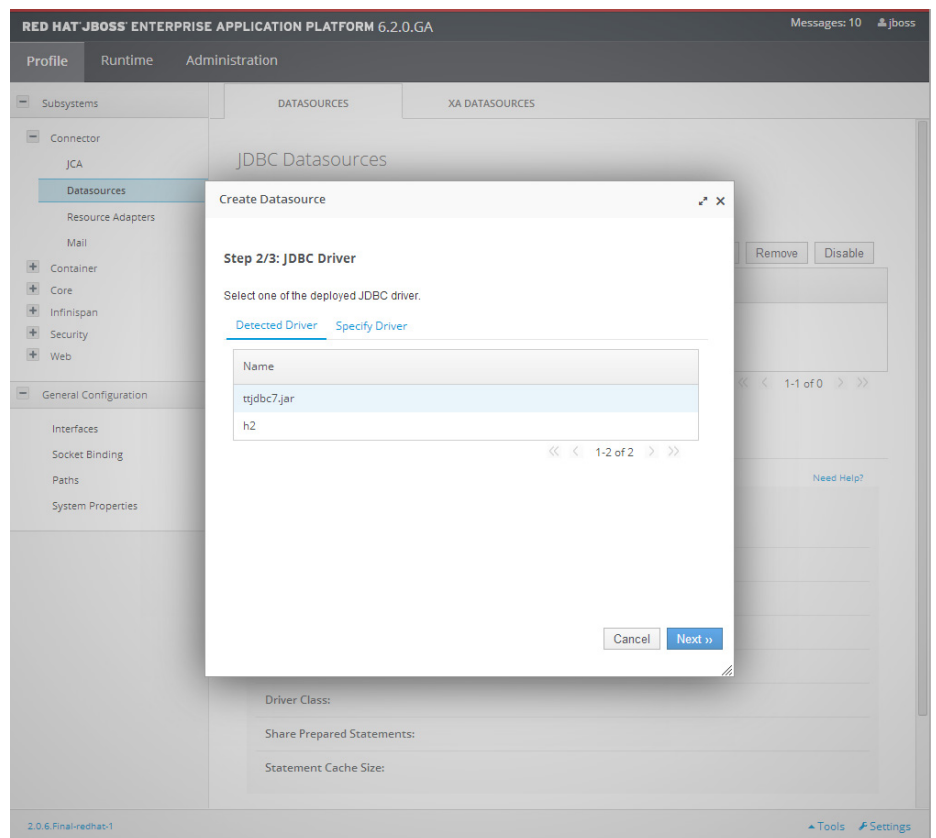


Figure 1.4 Step 2/3: JDBC Driver

8. Click **Next**.
9. In the Connection Settings step of the Create Datasource dialog, enter the TimesTen JDBC URL and TimesTen database user credentials for the connection. See Figure 1.5.

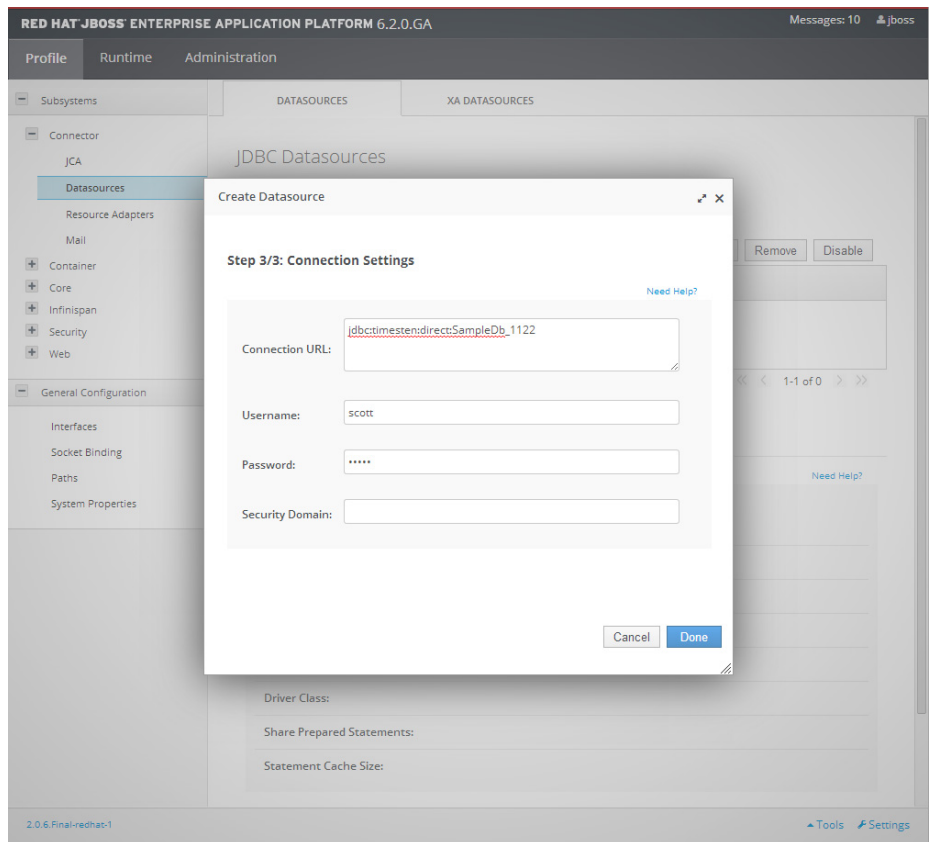


Figure 1.5 Step 3/3: Connection Settings

10. Click **Done**.
11. In the **DATASOURCES** tab, click **Enable**, and then click **Confirm** to enable the data source.
12. To verify that the data source can connect to TimesTen, click the **Connection** link, and then click **Test Connection**. See Figure 1.6.

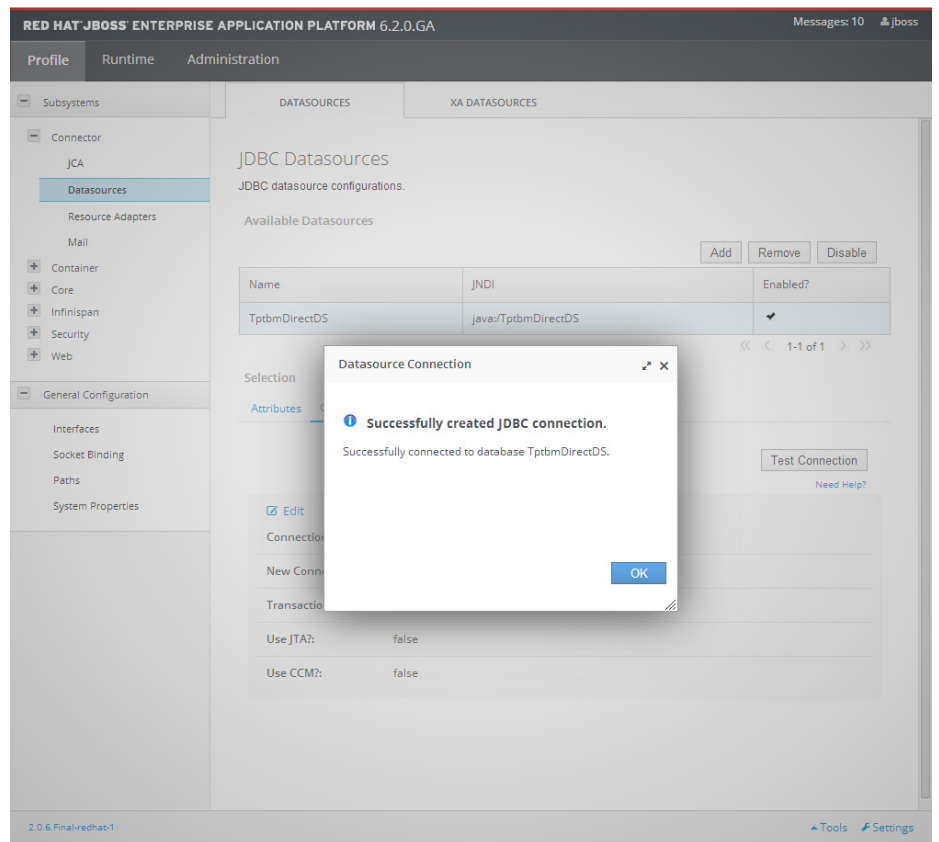


Figure 1.6 Datasource Connection dialog

Using JBoss JPA with TimesTen

JBoss server utilizes the Hibernate object-relational mapping framework to support the Java Persistence API (JPA). TimesTen supports Hibernate applications by providing a custom SQL dialect class called `TimesTenDialect1122`. This dialect class should be used for JBoss applications that connect to TimesTen using the Hibernate framework. For more information on the use of the TimesTen SQL dialect, see the “TimesTen for Hibernate 4” section in this document and the TimesTen Quick Start.

TIMESTEN FOR ORACLE WEBLOGIC SERVER 12.1

This section provides descriptions and examples of configuring the TimesTen JDBC driver for use with Oracle WebLogic Server 12.1.2.0. This section requires that you install both TimesTen and Oracle WebLogic Server on the same machine.

Configuring the TimesTen JDBC driver

Before you can use WebLogic Server to connect to TimesTen databases, the server environment must have access to the TimesTen JDBC driver jar file and the native TimesTen shared libraries.

1. In the environment where the server starts, set the `CLASSPATH` environment variable to include the TimesTen JDBC driver jar file. Use the Java 7 runtime version of the TimesTen driver jar file located at `tt_install_dir/lib/ttjdbc7.jar`. You can also set this environment variable within the domain's environment configuration script located at `wl_domain_dir/bin/setDomainEnv.sh` on UNIX systems or at `wl_domain_dir/bin/setDomainEnv.cmd` on Windows.
2. In the UNIX environment where the server starts, set the `LD_LIBRARY_PATH` (or equivalent variable for your OS) to include the TimesTen shared libraries located at `tt_install_dir/lib`. For servers on Windows platforms, append the `PATH` environment variable with the `tt_install_dir/bin` directory where TimesTen DLLs are located. You can also set these environment variables within the domain's environment configuration script located at `wl_domain_dir/bin/setDomainEnv.sh` on UNIX systems or at `wl_domain_dir/bin/setDomainEnv.cmd` on Windows.
3. Restart WebLogic Server for these changes to take effect.

Configuring TimesTen data sources

The TimesTen JDBC driver supports four types of connections to TimesTen databases:

- **Direct access with non-XA transactions** - This configuration provides the best database performance. The TimesTen database must reside on the same machine as WebLogic Server. The WebLogic Administration Console lists this driver type as Oracle TimesTen Direct Connection's Driver (Type 4).
- **Direct access with XA distributed transactions** - The driver uses JTA to support distributed transactions. The TimesTen database must reside on the same machine as WebLogic Server. The WebLogic Administration Console lists this driver type as Oracle TimesTen Direct Connection's Driver (Type 4 XA).

- **Client/server access with non-XA transactions** - The TimesTen database can reside on a remote machine that is running the TimesTen server. The WebLogic Administration Console lists this driver type as Oracle TimesTen Client Connection's Driver (Type 4).
- **Client/server access with XA distributed transactions** - The driver uses JTA to support distributed transactions. The TimesTen database can reside on a remote machine that is running the TimesTen server. The WebLogic Administration Console lists this driver type as Oracle TimesTen Client Connection's Driver (Type 4 XA).

Each type of TimesTen JDBC driver connection requires a unique WebLogic Server data source configuration. The next example references a TimesTen direct connection DSN named `SAMPLEDB_1122`. This DSN must be configured in the `.odbc.ini` or `sys.odbc.ini` file or the Windows Data Source Administrator before the example can work.

Use the next procedure as an example for configuring a TimesTen data source within the WebLogic Server Administration Console.

1. Connect to the WebLogic Server Administration Console.
2. In the Domain Structure tree, expand the **Services** node, and then click **Data Sources**.
3. Click **New**, and then select **Generic Data Source**.
4. Enter a name for the data source in the **Name** field.
5. Enter the JNDI name for the data source in the **JNDI Name** field.
6. In the Database Type drop-down list, select **Oracle TimesTen Direct Connection** for a direct connection DSN or **Oracle TimesTen Client Connection** for a client/server DSN. See Figure 2.1.

Home Log Out Preferences Record Help Welcome, weblogic Connected to: timesten_domain

Home > Summary of JDBC Data Sources > Summary of Services > Summary of JDBC Data Sources

Create a New JDBC Data Source

Back Next Finish Cancel

JDBC Data Source Properties

The following properties will be used to identify your new JDBC data source.
* Indicates required fields

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:

Back Next Finish Cancel

Figure 2.1 JDBC Data source properties

7. Click **Next**.
8. In the resulting screen, the Database Driver drop-down list will include two of the next four options:
 - Oracle TimesTen Direct Connection's Driver (Type 4)
 - Oracle TimesTen Direct Connection's Driver (Type 4 XA)
 - Oracle TimesTen Client Connection's Driver (Type 4)
 - Oracle TimesTen Client Connection's Driver (Type 4 XA)

Select one of these options based on application requirements. The non-XA versions of the TimesTen driver result in the best performance. Click **Next**.
9. At the Transaction Options screen, check the **Supports Global Transactions** checkbox. For the best performance, select the **One-Phase Commit** option. Note that if in the previous step you selected a XA version of the TimesTen driver, then these transaction options are not available. See Figure 2.2 for an example when using a non-XA driver type.

Home Log Out Preferences Record Help Welcome, weblogic Connected to: timesten_domain

Home > Summary of JDBC Data Sources > Summary of Services > Summary of JDBC Data Sources

Create a New JDBC Data Source

Back Next Finish Cancel

Transaction Options

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

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

Supports Global Transactions

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

Logging Last Resource

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

Emulate Two-Phase Commit

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

One-Phase Commit

Back Next Finish Cancel

Figure 2.2 Transaction Options

10. Click **Next**.
11. In the Connection Properties screen, enter a name in the **Database Name** field. You must use the name of the TimesTen DSN that the data source will connect to. This example uses the name SAMPLEDB_1122. The **Host Name** field and **Port** field are not relevant to TimesTen and may be left blank. Specify the TimesTen database user name in the **Database User Name** field and the TimesTen database user password in the **Password** and **Confirm Password** fields. See Figure 2.3.

Home Log Out Preferences Record Help Welcome, weblogic Connected to: timesten_domain

Home > Summary of JDBC Data Sources > Summary of Services > Summary of JDBC Data Sources

Create a New JDBC Data Source

Back Next Finish Cancel

Connection Properties

Define Connection Properties.

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

Database Name:

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

Host Name:

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

Port:

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:

Back Next Finish Cancel

Figure 2.3 Connection Properties

12. Click **Next**.
13. In the **Test Database Connection** screen, fill all available fields. Figure 2.4 shows an example with a DSN named `SAMPLEDB_1122` and a TimesTen database user called `scott`.

Home Log Out Preferences Record Help Welcome, weblogic Connected to: timesten_domain

Home > Summary of JDBC Data Sources > TptbmDS > Summary of JDBC Data Sources

Messages

✓ Connection test succeeded.

Create a New JDBC Data Source

Test Configuration Back Next Finish Cancel

Test Database Connection

Test the database availability and the connection properties you provided.

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

Driver Class Name: com.timesten.jdbc.Obser

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

URL: jdbc:timesten:direct:Sam

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

Database User Name: scott

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

Password: ●●●●●●●●●●●●●●●●

Confirm Password: ●●●●●●●●●●●●●●●●

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

Properties:
user=scott

Figure 2.4 Test Database Connection

14. Click **Test Configuration**.

A correct configuration results in the following message: “Connection test successful”. If the test fails, then review previous configuration steps. Common causes for failure include: CLASSPATH or LD_LIBRARY_PATH environment variables (or equivalent variable for your OS) configured incorrectly, or an incorrect TimesTen DSN name.

Click **Next**.

15. In the **Select Targets** screen, select the WebLogic servers to be associated with the data source. Click **Finish**.

16. The message “All changes have been activated. No restarts are necessary.” indicates that the configuration is complete.

Using WebLogic Server JPA with TimesTen

WebLogic Server supports the Java Persistence API (JPA) through the EclipseLink object-relational mapping framework. TimesTen supports EclipseLink applications. For additional information, see the “TimesTen for EclipseLink 2” section in this document and the TimesTen Quick Start.

TIMESTEN FOR ORACLE GLASSFISH SERVER 3.1

This section provides descriptions and examples of configuring the TimesTen JDBC driver for use with Oracle GlassFish Server 3.1. This section requires that you install both TimesTen and Oracle GlassFish Server on the same machine.

Configuring the TimesTen JDBC driver

Before you can use Oracle GlassFish Server to access TimesTen databases, the server environment must have access to the TimesTen JDBC driver jar file and the TimesTen shared libraries.

1. Copy the TimesTen JDBC driver jar file called `ttjdbc6.jar` from `tt_install_dir/lib` to the `gs_install_dir/glassfish/domains/gs_domain/lib` directory.
2. Set the `LD_LIBRARY_PATH` environment variable (or the equivalent variable for your OS) to include the `tt_install_dir/lib` directory. On Windows platforms, set the `PATH` environment variable to the `tt_install_dir/bin` directory where TimesTen DLLs are located. This variable must be set for the environment where the Oracle GlassFish Server runs.
3. Restart the Oracle GlassFish Server.

Configuring TimesTen connections

The TimesTen JDBC driver supports four different types of database connections:

- **Direct access with local transactions only** - This configuration provides the fastest database performance. The TimesTen database must reside on the same machine as Oracle GlassFish Server.
- **Direct access with XA distributed transaction support** - The driver uses JTA to support distributed transactions. The TimesTen database must reside on the same machine as Oracle GlassFish Server.
- **Client/server access with local transactions only** - The TimesTen database can reside on a remote machine that is running the TimesTen server.
- **Client/server access with XA distributed transaction support** - The driver uses JTA to support distributed transactions. The TimesTen database can reside on a remote machine that is running the TimesTen server.

Properties of TimesTen connections

To configure TimesTen connections in Oracle GlassFish Server use the next settings:

- **Direct access with local transactions only:**

TimesTen Class:
`com.timesten.jdbc.ObservableConnectionDS`

Example URL Property:
`jdbc:timesten:direct:SAMPLEDB_1122`

- **Direct access with XA distributed transaction support:**

TimesTen Class:
`com.timesten.jdbc.xa.TimesTenXADataSource`

Example URL Property:
`jdbc:timesten:direct:SAMPLEDB_1122`

- **Client/server access with local transactions only:**

TimesTen Class:
`com.timesten.jdbc.ObservableConnectionDS`

Example URL Property:
`jdbc:timesten:client:SAMPLEDBCS_1122`

- **Client/server access with XA distributed transaction support:**

TimesTen Class:
`com.timesten.jdbc.xa.TimesTenXADataSource`

Example URL Property:
`jdbc:timesten:client:SAMPLEDBCS_1122`

Configuring a TimesTen JDBC connection pool

To configure a TimesTen JDBC connection pool in Oracle GlassFish Server follow the next steps:

1. Connect to the Oracle GlassFish Server administration console.
2. In the left pane, expand **Resources**, then expand **JDBC**, and select **JDBC Connection Pools**.
3. Click **New**.
4. In the New JDBC Connection Pool (Step 1 of 2) screen, enter a name for the new pool in the **Name** field. In the Resource Type drop-down list, select `javax.sql.ConnectionPoolDataSource`, if the application requires only local transaction support. If the application requires XA distributed transaction support, then select

`javax.sql.XADataSource`. In the **Database Driver Vendor** drop-down list, select 'Oracle'. See Figure 3.1.

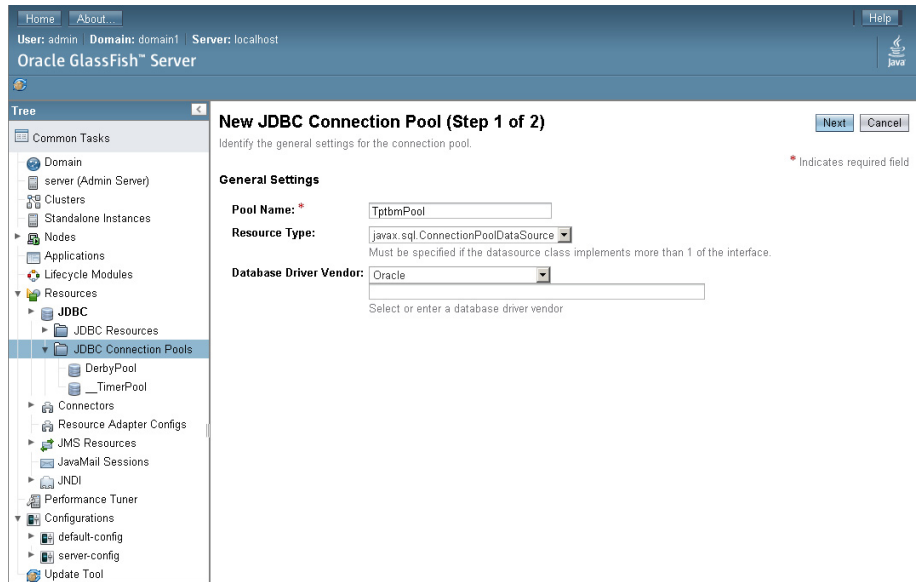


Figure 3.1 New JDBC Connection Pool

5. Click **Next**.
6. In the New Connection Pool (Step 2 of 2) screen, enter `com.timesten.jdbc.ObservableConnectionDS` in the **Datasource Classname** field for local transactions. Enter `com.timesten.jdbc.xa.TimesTenXADataSource` for XA distributed transactions.

Scroll down to the Additional Properties section at the bottom of the screen, and click **Add Property**. Select the checkbox in the new row that appears. Type 'url' in the **Name** column. In the **Value** column, enter the URL for the TimesTen database associated with the connection pool. A TimesTen URL takes the form:

`jdbc:timesten:[direct|client]:DSN`. Specify the name of the TimesTen database user for the connection in the **Value** field of the 'user' property. Specify the TimesTen user password in the **Value** field of the 'password' property. Figure 3.2 shows an example for a connection to a direct DSN named `SAMPLEDB_1122` using the TimesTen user account `scott/tiger`.

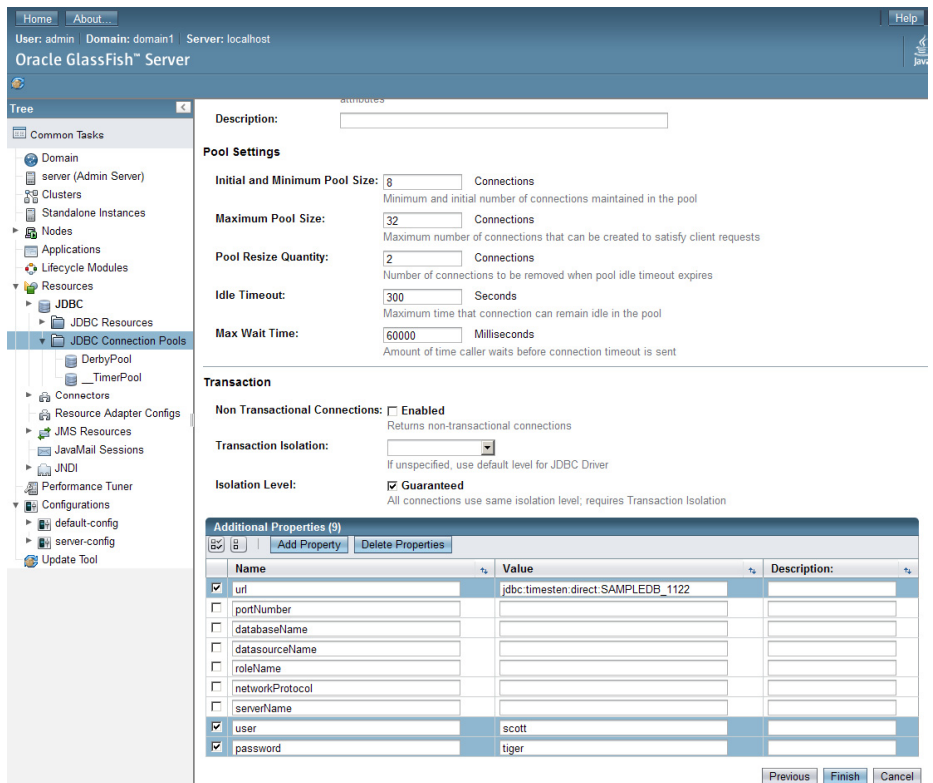


Figure 3.2 Additional Properties

7. Click **Finish**. The TimesTen connection pool configuration is now complete.

Configuring a TimesTen JDBC resource

You must create a JDBC resource and associate it to the TimesTen connection pool for applications to access the TimesTen database. To perform this task, follow the next steps:

1. Connect to the Oracle GlassFish Server administration console.
2. In the left pane, expand **Resources**, then expand **JDBC**, and expand **JDBC Resources**.
3. Click **New**.
4. In the **New JDBC Resource** screen, enter a JNDI name for the resource. In the **Pool Name** field, select the name of the TimesTen connection pool you previously created. Figure 3.3 shows an example for a TimesTen connection pool called `TptbmPool`.



Figure 3.3 New JDBC Resource

5. Click **OK**. The TimesTen database configuration for Oracle GlassFish Server is now complete.

Using Oracle GlassFish Server JPA with TimesTen

Oracle GlassFish Server uses the EclipseLink object-relational mapping framework to support the Java Persistence API (JPA). TimesTen supports EclipseLink applications. For additional information, see the “TimesTen for EclipseLink 2” section in this document and the TimesTen Quick Start.

TIMESTEN FOR WEBSHERE APPLICATION SERVER 8.5

This section provides descriptions and examples of configuring the TimesTen JDBC driver for use with IBM WebSphere Application Server 8.5.5.1. This section requires that you install both TimesTen and WebSphere Application Server on the same machine.

Configuring TimesTen connections

The TimesTen JDBC driver supports four different types of database connections.

- **Direct access with local transactions only** - This configuration provides the fastest database performance. The TimesTen database must reside on the same machine as WebSphere Application Server.
- **Direct access with XA distributed transaction support** - The driver uses JTA to support distributed transactions. The TimesTen database must reside on the same machine as WebSphere Application Server.
- **Client/server access with local transactions only** - The TimesTen database can reside on a remote machine that is running the TimesTen server.
- **Client/server access with XA distributed transaction support** - The driver uses JTA to support distributed transactions. The TimesTen database can reside on a remote machine that is running the TimesTen server.

Properties of TimesTen connections

To configure TimesTen connections in WebSphere use the next settings:

- **Direct access with local transactions only:**

TimesTen Class:
`com.timesten.jdbc.ObservableConnectionDS`

Example URL Property:
`jdbc:timesten:direct:SAMPLEDB_1122`

- **Direct access with XA distributed transaction support:**

TimesTen Class:
`com.timesten.jdbc.xa.TimesTenXADataSource`

Example URL Property:
`jdbc:timesten:direct:SAMPLEDB_1122`

- **Client/server access with local transactions only:**

TimesTen Class:
`com.timesten.jdbc.ObservableConnectionDS`

Example URL Property:
`jdbc:timesten:client:SAMPLEDBCS_1122`

- **Client/server access with XA distributed transaction support:**

TimesTen Class:
`com.timesten.jdbc.xa.TimesTenXADataSource`

Example URL Property:
`jdbc:timesten:client:SAMPLEDBCS_1122`

Configuring a TimesTen JDBC provider

Before you can use WebSphere to access TimesTen databases, you must set the server environment to have access to the TimesTen JDBC driver jar file and the native shared libraries. In WebSphere, you can do this by configuring a JDBC provider in the WebSphere administration console. Follow the next steps:

1. Connect to the WebSphere administrative console.
2. In the left pane, expand **Resources**, then expand **JDBC**, and then select **JDBC Providers**.
3. Select a **Scope** from the drop-down list.
4. Click **New**.
5. In the Create new JDBC provider screen, select **User-defined** for **Database type**. If the application requires distributed XA transaction support, enter `com.timesten.jdbc.xa.TimesTenXADataSource` in the **Implementation class name** field. Otherwise, enter `com.timesten.jdbc.ObservableConnectionDS`. Enter a name for the configuration in the **Name** field. See Figure 4.1.

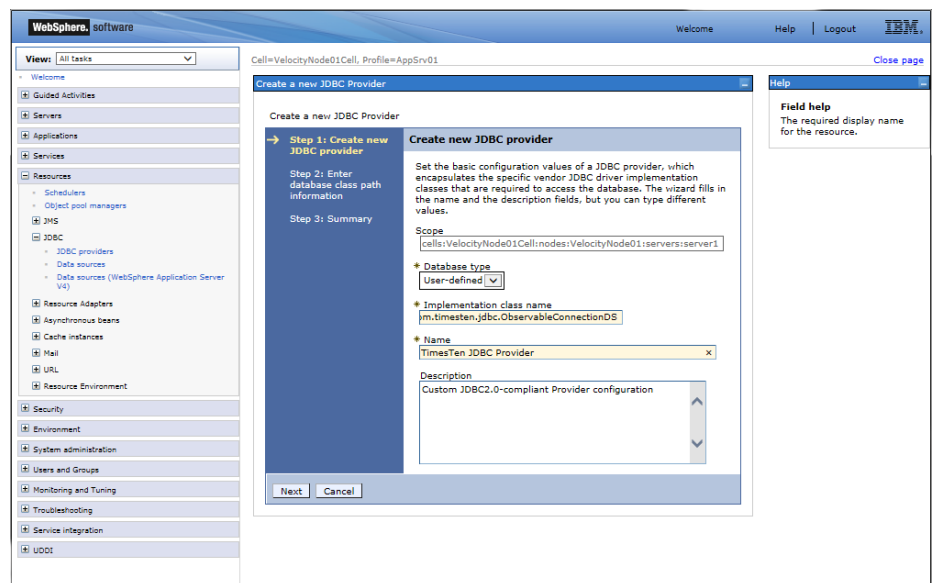


Figure 4.1 Create new JDBC provider

Click **Next**.

6. In the Enter database class path information screen, enter the path to the TimesTen JDBC driver located at `tt_install_dir/lib/ttjdbc6.jar`. See Figure 4.2.

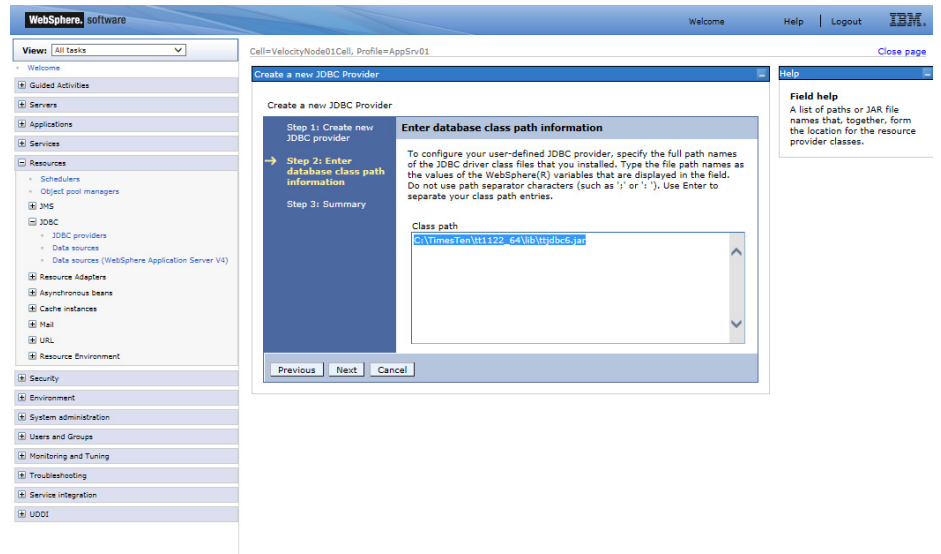


Figure 4.2 Enter database class path information

Click **Next**.

7. In the summary screen, click **Finish**.
8. In the JDBC providers screen, click the TimesTen provider name that you just created.
9. In the General Properties section of Configuration tab, enter the path to the TimesTen shared libraries in the **Native library path** field. On UNIX platforms, the path is `tt_install_dir/lib`. On Windows platforms, the path is `tt_install_dir/bin`. See Figure 4.3.

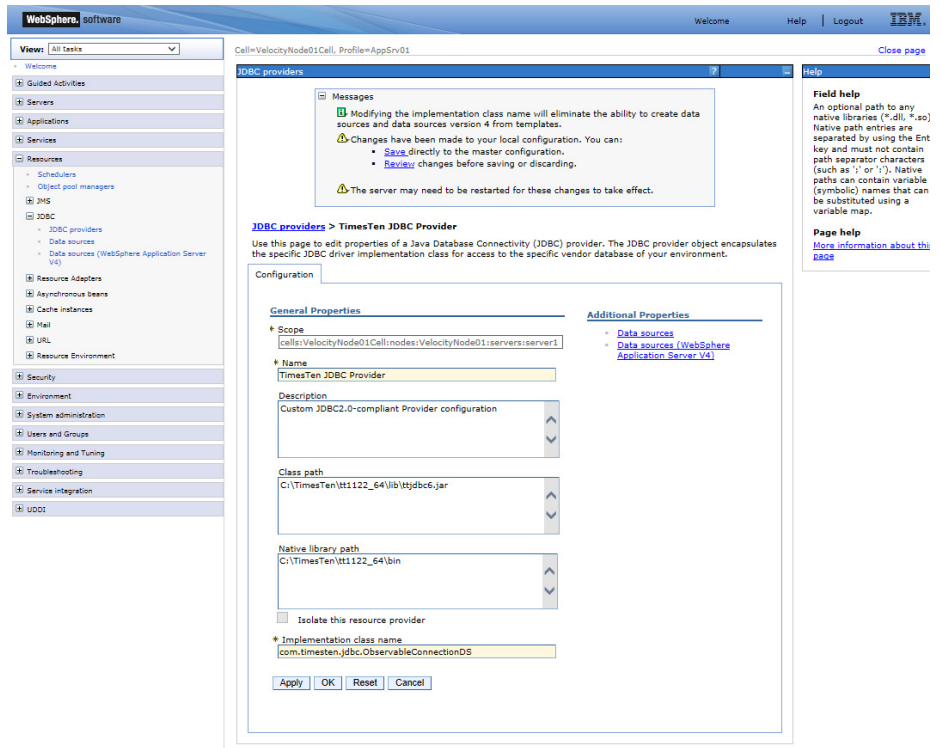


Figure 4.3 General properties

Click **Apply**.

10. Click **Save** to save the new configuration.

Configuring TimesTen data sources

WebSphere data sources are associated to the specific JDBC providers defined in the previous section. Follow these steps to configure TimesTen data sources from the WebSphere administration console.

1. Connect to the WebSphere administration console.
2. In the left pane, expand **Resources**, then expand **JDBC**, and then select **Data sources**.
3. Select a **Scope** from the drop-down list.
4. Click **New**.
5. In the Enter basic data source information screen, enter a name and a JNDI name for the data source. Click **Next**.
6. In the Select JDBC provider screen, select the TimesTen JDBC Provider you created in the previous section. Click **Next**.
7. In the Enter database specific properties for the data source screen, use `com.ibm.websphere.rsadapter.GenericDataStoreHelp`

r for the **Data store helper class name** field. Make sure that the **Use this database in container managed persistence (CMP)** checkbox is checked. Click **Next**.

8. In the Setup security aliases screen, click **Next**.
9. In the following summary screen, click **Finish**.
10. In the Data sources screen, click on the name of the TimesTen data source that you just created.
11. Click the **Custom properties** link.
12. Click the **url** property. (If the property does not exist, then create it by clicking **New**.)
13. In the **value** field, enter the TimesTen JDBC URL for the DSN that this data source should connect to. A TimesTen URL takes the form:
`jdbc:timesten:<direct|client>:<DSN>`. Click **OK**.
14. Click the **user** property, and enter the TimesTen database user name for the connection in the **value** field. Click **OK**.
15. Click the **password** property, and enter the TimesTen database password for the connection in the **value** field. Click **OK**.
16. Click the **webSphereDefaultIsolationLevel** property and enter '2' in the **value** field. Click **OK**.

Figure 4.4 shows an example for a direct DSN.

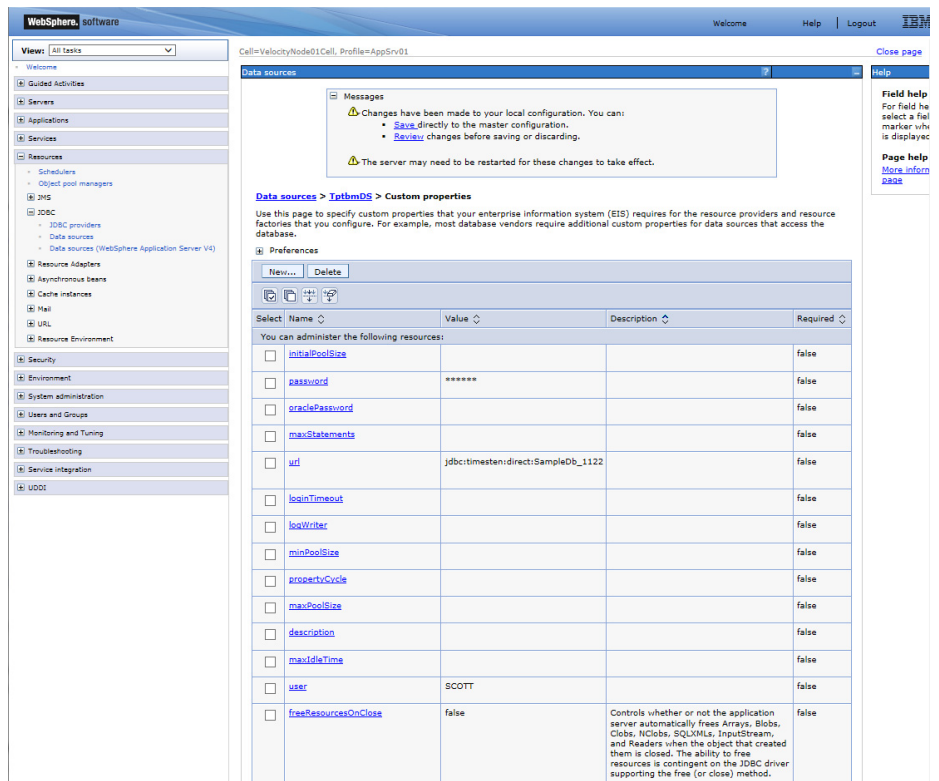


Figure 4.4 Data Sources Custom Properties

17. Click the **Save** link near the top of the screen to save the new configuration.
18. Restart the WebSphere server before attempting to make connections to the new TimesTen data source.

Using WebSphere Application Server JPA with TimesTen

WebSphere Server supports the Java Persistence API (JPA) using the OpenJPA object-relational mapping framework. TimesTen supports OpenJPA applications. For additional information, see the “TimesTen for OpenJPA 2” section in this document and the TimesTen Quick Start.

TIMESTEN FOR ECLIPSELINK 2

This section provides descriptions and examples of configuring the TimesTen JDBC driver for use with EclipseLink 2.5.1 JPA. EclipseLink is a further development of the Oracle TopLink object-relational mapping framework. This section requires that you install both TimesTen and EclipseLink on the same machine.

Configuring the TimesTen JDBC driver for EclipseLink

To access a TimesTen database from an EclipseLink application, you must set the CLASSPATH environment variable to include the TimesTen JDBC driver jar file. When using the Java 6 runtime, add `tt_install_dir/lib/ttjdbc6.jar` to the CLASSPATH. When using the Java 7 runtime, add the `tt_install_dir/lib/ttjdbc7.jar` file to the CLASSPATH.

Set the LD_LIBRARY_PATH (or equivalent for your OS) environment variable to include the path to the TimesTen shared libraries located at `tt_install_dir/lib`. On Windows platforms, the PATH environment variable should be set to the `tt_install_dir/bin` directory where TimesTen DLLs are located.

The EclipseLink distribution includes a platform class called `org.eclipse.persistence.platform.database.TimesTen7Platform`. This is the recommended platform class for TimesTen use with EclipseLink applications. An example `persistence-eclipselink2.xml` configuration file for EclipseLink JPA and TimesTen is located under the `quickstart` directory in your TimesTen installation.

Example `persistence.xml` configuration file for EclipseLink

```
<?xml version="1.0" encoding="UTF-8"?>

<persistence xmlns="http://java.sun.com/xml/ns/persistence"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/persistence
http://java.sun.com/xml/ns/persistence/persistence_1_0.xsd"
  version="1.0">

  <persistence-unit name="TptbmEclipseLink" transaction-
type="RESOURCE_LOCAL">

    <provider>org.eclipse.persistence.jpa.PersistenceProvider</provider>
    <class>com.timesten.tptbmas.Tptbm</class>

    <properties>
      <property name="eclipseLink.logging.level" value="INFO"/>
      <property name="eclipseLink.target-database"
value="org.eclipse.persistence.platform.database.TimesTen7Platform"/>
      <property name="javax.persistence.jdbc.driver"
value="com.timesten.jdbc.TimesTenDriver"/>
      <property name="javax.persistence.jdbc.url"
value="jdbc:timesten:direct:SAMPLEDB_1122"/>
      <property name="javax.persistence.jdbc.password" value="tiger"/>
      <property name="javax.persistence.jdbc.user" value="SCOTT"/>
    </properties>

  </persistence-unit>
</persistence>
```

TIMESTEN FOR HIBERNATE 4

This section provides descriptions and examples of configuring the TimesTen JDBC driver for use with Hibernate 4. This section requires that you install both TimesTen and Hibernate on the same machine.

Configuring the TimesTen JDBC driver for Hibernate

To access a TimesTen database from a Hibernate application, you must set the CLASSPATH environment variable to include the TimesTen JDBC driver jar file. When using the Java 6 runtime, add `tt_install_dir/lib/ttjdbc6.jar` to the CLASSPATH. When using the Java 7 runtime, add the `tt_install_dir/lib/ttjdbc7.jar` file to the CLASSPATH.

Set the LD_LIBRARY_PATH (or equivalent for your OS) environment variable to include the path to the TimesTen shared libraries located at `tt_install_dir/lib`. On Windows platforms, the PATH environment variable should be set to the `tt_install_dir/bin` directory where TimesTen DLLs are located.

A Hibernate SQL dialect class called `org.hibernate.dialect.TimesTenDialect1122` has been developed for use with TimesTen. This dialect is optimized for the latest SQL features of TimesTen, and it is the recommended dialect for TimesTen use with Hibernate applications. This dialect is not included in current versions of the Hibernate distribution. The dialect is available as part of the TimesTen Quick Start example programs that are included as an optional component of a TimesTen installation. The java source code for the dialect is located in the `tt_install_dir/quickstart/sample_code/orm/config/hibernate4` directory of the TimesTen installation. This directory includes an Ant build script and instructions for compiling and using the dialect in TimesTen in conjunction with Hibernate applications.

Hibernate configuration properties for TimesTen

Hibernate uses properties defined in a `hibernate.properties` file or a `hibernate.cfg.xml` configuration file to determine which JDBC driver and which SQL dialect class to use for a session.

When using TimesTen with Hibernate, set the next property values:

```
hibernate.dialect=org.hibernate.dialect.TimesTenDialect1122
hibernate.connection.driver_class=com.timesten.jdbc.TimesTenDriver
hibernate.connection.url=jdbc:timesten:<direct|client>:<DSN>
```

The following property values are recommended when using TimesTen with Hibernate:

```
hibernate.connection.isolation=2
hibernate.jdbc.use_get_generated_keys=false
hibernate.jdbc.use_scrollable_resultset=false
hibernate.jdbc.batch_size=256
```

An example `hibernate.cfg.xml` configuration file is located under the `quickstart` directory in your TimesTen installation.

Example of `hibernate.cfg.xml` configuration file

```
<?xml version='1.0' encoding='utf-8'?>
<!DOCTYPE hibernate-configuration PUBLIC
    "-//Hibernate/Hibernate Configuration DTD//EN"
    "http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">
<hibernate-configuration>

    <!-- a SessionFactory instance -->
    <session-factory name="Tptbm">

        <!-- properties -->
        <property name="hibernate.connection.url">
            jdbc:timesten:direct:SAMPLEDB_1122
        </property>
        <property name="hibernate.connection.username">SCOTT</property>
        <property name="hibernate.connection.password">tiger</property>
        <property name="hibernate.connection.driver_class">
            com.timesten.jdbc.TimesTenDriver
        </property>
        <property name="hibernate.dialect">
            org.hibernate.dialect.TimesTenDialect1122
        </property>
        <!-- Connection.TRANSACTION_READ_COMMITTED = 2 -->
        <property name="hibernate.connection.isolation">2</property>
        <property name="hibernate.jdbc.fetch_size">32</property>
        <property name="hibernate.jdbc.batch_size">256</property>
        <property name="hibernate.jdbc.batch_versioned_data">
            True
        </property>
        <property name="hibernate.jdbc.use_streams_for_binary">
            False
        </property>
        <property name="hibernate.jdbc.use_get_generated_keys">
            False
        </property>
        <property name="hibernate.jdbc.use_scrollable_resultset">
            False
        </property>
        <property name="hibernate.cache.use_query_cache">
            False
        </property>
        <property name="hibernate.cache.use_second_level_cache">
            False
        </property>

        <property name="hibernate.show_sql">>false</property>
        <property name="hibernate.connection.pool_size">4</property>

        <!-- mapping file -->
        <mapping resource="META-INF/Tptbm.hbm.xml" />

    </session-factory>
```

```
</hibernate-configuration>
```

When using Hibernate with the JPA API, the same TimesTen properties can be specified in the `persistence.xml` configuration file. An example JPA configuration file (`persistence-hibernate.xml`) for a TimesTen Hibernate application is located under the `quickstart` directory in your TimesTen installation.

Example of the `persistence.xml` configuration file

```
<?xml version="1.0" encoding="UTF-8"?>

<persistence xmlns="http://java.sun.com/xml/ns/persistence"
             xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
             xsi:schemaLocation="http://java.sun.com/xml/ns/persistence
http://java.sun.com/xml/ns/persistence/persistence_1_0.xsd"
             version="1.0">

  <persistence-unit name="TptbmHibernate" transaction-
type="RESOURCE_LOCAL">

    <provider>org.hibernate.ejb.HibernatePersistence</provider>
    <class>com.timesten.tptbmas.Tptbm</class>

    <properties>
      <property name="hibernate.connection.url"
                value="jdbc:timesten:direct:SAMPLEDB_1122" />

      <property name="hibernate.connection.username" value="SCOTT"/>
      <property name="hibernate.connection.password" value="tiger"/>

      <property name="hibernate.connection.driver_class"
                value="com.timesten.jdbc.TimesTenDriver"/>

      <property name="hibernate.dialect"
                value="org.hibernate.dialect.TimesTenDialect1122"/>

      <!-- Connection.TRANSACTION_READ_COMMITTED = 2 -->
      <property name="hibernate.connection.isolation" value="2"/>

      <property name="hibernate.jdbc.fetch_size" value="32"/>
      <property name="hibernate.jdbc.batch_size" value="256"/>
      <property name="hibernate.jdbc.batch_versioned_data"
value="true"/>

      <property name="hibernate.jdbc.use_streams_for_binary"
                value="false"/>
      <property name="hibernate.jdbc.use_get_generated_keys"
                value="false"/>
      <property name="hibernate.jdbc.use_scrollable_resultset"
                value="false"/>

      <property name="hibernate.cache.use_query_cache" value="false"/>
      <property name="hibernate.cache.use_second_level_cache"
                value="false"/>

      <property name="hibernate.show_sql" value="false"/>
      <property name="hibernate.connection.pool_size" value="4"/>
    </properties>
  </persistence-unit>
</persistence>
```

TIMESTEN FOR OPENJPA 2

This section provides descriptions and examples of configuring the TimesTen JDBC driver for use with OpenJPA 2.2.2. This section requires that you install both TimesTen and OpenJPA on the same machine.

Configuring the TimesTen JDBC driver for OpenJPA

When accessing a TimesTen database from an OpenJPA application, you must set the CLASSPATH environment variable to include the TimesTen JDBC driver jar file. When using the Java 6 runtime, add `tt_install_dir/lib/ttjdbc6.jar` to the CLASSPATH. When using the Java 7 runtime, add the `tt_install_dir/lib/ttjdbc7.jar` file to the CLASSPATH.

Set the LD_LIBRARY_PATH (or equivalent for your OS) environment variable to include the path to the TimesTen shared libraries located at `tt_install_dir/lib`. On Windows platforms, the PATH environment variable should be set to the `tt_install_dir/bin` directory where TimesTen DLLs are located.

You can configure OpenJPA with a property called `openjpa.jdbc.DBDictionary`, which defines a SQL dialect for a database provider. For TimesTen connections, the recommended value for this property is 'oracle'. An example `persistence-openjpa2.xml` configuration file for OpenJPA and TimesTen is located under the `quickstart` directory in your TimesTen installation.

Example persistence.xml configuration file for OpenJPA

```
<?xml version="1.0" encoding="UTF-8"?>
<persistence xmlns="http://java.sun.com/xml/ns/persistence"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/persistence
http://java.sun.com/xml/ns/persistence/persistence_1_0.xsd"
  version="1.0">
  <persistence-unit name="TptbmOpenJPA" transaction-
type="RESOURCE_LOCAL">
<provider>org.apache.openjpa.persistence.PersistenceProviderImpl</provid
er>
  <class>com.timesten.tptbmas.Tptbm</class>
  <class>com.timesten.tptbmas.TptbmPKey</class>
  <properties>
    <property name="openjpa.ConnectionUserName" value="SCOTT" />
    <property name="openjpa.ConnectionPassword" value="tiger" />
    <property name="openjpa.ConnectionURL"
value="jdbc:timesten:direct:SAMPLEDB_1122" />
    <property name="openjpa.ConnectionDriverName"
value="com.timesten.jdbc.TimesTenDriver" />
    <property name="openjpa.jdbc.DBDictionary" value="oracle" />
    <property name="openjpa.ConnectionRetainMode" value="always" />
    <property name="openjpa.Multithreaded" value="true" />
  </properties>
</persistence-unit>
</persistence>
```



Configuring Oracle TimesTen In-Memory Database 11.2.2 for Application Servers and Object-Relational Mapping Frameworks
April 2014

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com

Copyright © 2011 - 2014, Oracle. All rights reserved.
This document is provided for information purposes only and the contents hereof are subject to change without notice.
This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission. Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.