Oracle® SQL Developer

Oracle TimesTen In-Memory Database Support User's Guide

Release 18.2

E71314-03

July 2018

ORACLE®

Oracle SQL Developer Oracle TimesTen In-Memory Database Support User's Guide, Release 18.2

E71314-03

Contents

3 Connecting to a TimesTen Database

4 Working with TimesTen Objects

12 Working with TimesTen Replication

13 Working with TimesTen Performance Metrics

14 TimesTen Reports

Index

Preface

Oracle SQL Developer is a graphical user interface (GUI) tool that gives database developers a convenient way to perform basic tasks. You can browse, create, edit and drop particular database objects; run SQL statements and scripts; manipulate and export data; and view and create reports. This guide is a supplement to the *Oracle SQL Developer User's Guide* for TimesTen In-Memory Database specific features.

Oracle TimesTen In-Memory Database (TimesTen) is a relational database that is memory-optimized for fast response and throughput. The database resides entirely in memory at runtime and is persisted to disk storage.

- Oracle TimesTen In-Memory Database in classic mode, or TimesTen Classic, refers to single-instance and replicated databases (as in previous releases).

- Oracle TimesTen In-Memory Database in grid mode, or TimesTen Scaleout, refers to multiple-instance distributed databases. TimesTen Scaleout is a grid of interconnected hosts running instances that work together to provide fast access, fault tolerance, and high availability for in-memory data. A grid contains one or more databases and each database is distributed across all instances of the grid.

- TimesTen alone refers to both classic and grid modes (such as in references to TimesTen utilities, releases, distributions, installations, actions taken by the database, and functionality within the database).

- TimesTen Application-Tier Database Cache, or TimesTen Cache, is an Oracle Database Enterprise Edition option. TimesTen Cache is ideal for caching performance-critical subsets of an Oracle database into cache tables within TimesTen databases for improved response time in the application tier. Cache tables can be read-only or updatable. Applications read and update the cache tables using standard Structured Query Language (SQL) while data synchronization between the TimesTen database and the Oracle database is performed automatically. TimesTen Cache offers all of the functionality and performance of TimesTen Classic, plus the additional functionality for caching Oracle Database tables.

- TimesTen Replication features, available with TimesTen Classic or TimesTen Cache, enable high availability.

You can use Oracle SQL Developer to connect to a target TimesTen database and perform operations on particular objects in the database.

This guide describes the various tasks that you can perform with SQL Developer for TimesTen databases. However, TimesTen Scaleout supports and includes most of the features of TimesTen Classic database; it does not support any of the features of the TimesTen Application-Tier Database Cache. For more information on what features are

not supported in TimesTen Scaleout, see "Comparison between TimesTen Scaleout and TimesTen Classic" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

Audience

This guide is intended for those who use the Oracle SQL Developer tool with TimesTen In-Memory Database, TimesTen Application-Tier Database Cache, or TimesTen Replication. Application developers and system administrators can use SQL Developer to create and manage cache groups, and other database objects.

Related documents

TimesTen documentation is available at `https://docs.oracle.com/database/timesten-18.1` and on the Oracle Technology Network:

`http://www.oracle.com/technetwork/database/database-technologies/timesten/documentation/index.html`

Oracle Database documentation is also available on the Oracle documentation website. This may be especially useful for Oracle Database features that TimesTen supports but does not attempt to fully document, such as OCI and Pro*C/C++.

The SQL Developer software download is available on Oracle Technology Network (OTN) at:

`http://www.oracle.com/technetwork/developer-tools/sql-developer/overview/index.html`

Conventions

TimesTen supports multiple platforms. Unless otherwise indicated, the information in this guide applies to all supported platforms. The term Windows refers to all supported Windows platforms. The term UNIX applies to all supported UNIX platforms. The term Linux is used separately. See "Platforms" in *Oracle TimesTen In-Memory Database Release Notes* for specific platform versions supported by TimesTen.

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

In addition, TimesTen documentation uses the following special conventions:

Convention	Meaning
installation_dir	The path that represents the directory where TimesTen is installed.
timesten_home	The path that represents the home directory of a TimesTen instance.

Convention	Meaning
release or *rr*	The first three parts in a release number with or without dots. The first three parts of a release number represent a major TimesTen release. For example, 181 or 18.1 represents TimesTen 12*c* Release 12.1.1.
jdk_ver	One or two digits that represent the version number of a major JDK release. For example, 14 is for JDK 1.4 and 5 is for JDK 5.0.
DSN	TimesTen data source name (for the TimesTen database).

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Technical support

For information about obtaining technical support for TimesTen products, go to the following Web address:

http://www.oracle.com/support/contact.html

What's New

This section summarizes the new features of TimesTen In-Memory Database supported in Oracle SQL Developer Release 18.2. There are some features that are only supported in TimesTen Scaleout. If a feature is only supported in TimesTen Scaleout, it will be noted.

> **Note:** In this document:
>
> - Oracle TimesTen In-Memory Database in classic mode, or TimesTen Classic, refers to single-instance and replicated databases (as in previous releases)..
>
> - Oracle TimesTen In-Memory Database in grid mode, or TimesTen Scaleout, refers to multiple-instance distributed databases. TimesTen Scaleout is a grid of interconnected hosts running instances that work together to provide fast access, fault tolerance, and high availability for in-memory data. A grid contains one or more databases and each database is distributed across all instances of the grid.
>
> - TimesTen alone refers to both classic and grid modes (such as in references to TimesTen utilities, releases, distributions, installations, actions taken by the database, and functionality within the database).

New features for release of SQL Developer 18.2

- You can alter a table that is already created. See "Edit a table" on page 4-8 for more information.

Features supported in TimesTen In-Memory Database Release 18.1 or later:

- You can use SQL Developer to work with a TimesTen Scaleout grid and TimesTen Scaleout databases. See "Working with TimesTen Scaleout" on page 2-1 for more information. For more information on TimesTen Scaleout, see *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

New features for release of SQL Developer 4.0

- When you create a named connection to connect to a TimesTen database, the **User specified** connection type is now called **Advanced**.

- You can set the **Autocommit** option directly from the **New/Select Database Connection** dialog. This new autocommit option is set at the connection level. In previous releases the autocommit option is set at the running instance level.

- There is a new **Easy Client/Server** connection type to connect to a TimesTen database. This new connection type allows you to connect to local and remote TimesTen databases without having to define a client DSN. For more information, see "Easy Client/Server" on page 3-4.

Features supported in TimesTen Release 11.2.2.5 or later:

- You can capture `ttStats` snapshots and generate `ttStats` reports that compare two snapshots. A `ttStats` snapshot is a collection of performance metrics. TimesTen collects performance metrics from TimesTen system tables, system views, and built-in procedures. For more information, see "Working with TimesTen Performance Metrics" on page 13-1.

- There are new pre-defined Performance reports. Performance reports are reports that show statistics information for the TimesTen database and statistics snapshots of the database. These reports are available in the **Performance** category of the **TimesTen Reports**. For more information, see "Performance" on page 14-3.

Features supported in TimesTen Release 11.2.2.4 or later:

- SQL Developer supports the TimesTen Index Advisor which can evaluate a SQL workload and recommend indexes. The indexes that the TimesTen Index Advisor recommends can improve the performance for the following: joins, single table scans, and `ORDER BY` or `GROUP BY` operations. For more information, see "Working with the TimesTen Index Advisor" on page 6-1.

- You can load data using parallel threads from an Oracle database into a TimesTen database without creating cache groups or cache tables. For more information, see "Loading data from an Oracle database into a TimesTen database" on page 7-1.

1

Prerequisites to using SQL Developer

Before you use SQL Developer with TimesTen or TimesTen In-Memory Database, you must perform the tasks:

- Setting environment variables for TimesTen and SQL Developer

For information about installing and starting SQL Developer, see *Oracle SQL Developer User's Guide*.

Setting environment variables for TimesTen and SQL Developer

To access a TimesTen database from SQL Developer, install SQL Developer on the same system as the TimesTen Data Manager or Client. If you use SQL Developer to access local databases, install SQL Developer on the same system as the TimesTen Data Manager. If you use SQL Developer to access remote databases, install SQL Developer on the same system as the TimesTen Client.

Before you start SQL Developer, execute the `installation_dir`/bin/ttenv.sh or `installation_dir`/bin/ttenv.csh shell script on Linux and UNIX systems, the `installation_dir`\bin\ttenv.bat batch file on Microsoft Windows systems, or the `installation_dir`/bin/ttenv.sh on Mac OS X. The shell script or batch file adds or modifies the environment variables:

- CLASSPATH includes the TimesTen `installation_dir`/lib/ttjdbc8.jar file on Linux and Mac OS X systems or the TimesTen `installation_dir`\lib\ttjdbc8.jar file on Microsoft Windows systems. SQL Developer Release 18.2 requires that you have installed Java Development Kit (JDK) 8.

  To determine the third party JDBC drivers SQL Developer is using, from the **Tools** menu, select **Preferences**. In the **Preferences** dialog, click the **+** to the left of the **Database** node. Select **Third Party JDBC Drivers**. View the path name of all third party JDBC drivers including the TimesTen JDBC drivers.

- LD_LIBRARY_PATH includes the TimesTen `installation_dir`/lib directory on Linux systems.

- PATH includes the TimesTen `installation_dir`/bin directory on Linux and Mac OS X systems or the TimesTen `installation_dir`\bin directory on Microsoft Windows systems.

Additional prerequisites for specific platforms:

- Additional prerequisites for Microsoft Windows
- Additional prerequisites for macOS

Additional prerequisites for Microsoft Windows

If you are using a TimesTen client version earlier than 18.1, you need to enable TimesTen Scaleout for SQL Developer:

1. Edit the SQL Developer `product.conf` file. In most cases the file is located in `C:\Users\`*`user`*`\AppData\Roaming\sqldeveloper\`*`release`*`\`

   For example, if you are using release 18.2 of SQL Developer, the release directory is `18.2.0`.

2. Add the following setting at the end of the `product.conf` file:

   ```
   AddVMOption
   -Doracle.dbtools.raptor.timesten.grid.internal.relaxDriverVersionCheck=true
   ```

Additional prerequisites for macOS

To use SQL Developer with TimesTen on macOS, perform these prerequisites:

- Enable TimesTen client functionality
- Create script that sources TimesTen environment variables

Enable TimesTen client functionality

For macOS, you need to edit the SQL Developer product.conf file to enable TimesTen client functionality:

1. Edit the SQL Developer `product.conf` file. In most cases the file is located in `Users/`*`user`*`/.sqldeveloper/`*`release`*`/`

   For example, if you are using release 18.2 of SQL Developer, the release directory is `18.2.0`.

2. Add the following setting at the end of the `product.conf` file:

   > **Note:** Ensure that you replace *`timesten_home`* with the home directory of your local TimesTen client instance.

   ```
   AddVMOption -Djava.library.path=/timesten_home/install/lib
   ```

3. Save the changes to the `product.conf` file.

Create script that sources TimesTen environment variables

It is recommended that you create a shell script file that sources the TimesTen environment variables whenever you start SQL Developer.

1. Create a shell script file, `env.sh`, in the `/Users/`*`user`*`/.sqldeveloper` directory.

2. Add the following to the `env.sh` file:

   > **Note:** Ensure that you replace *`timesten_home`* with the home directory of your local TimesTen client instance.

   ```
   source timesten_home/bin/ttenv.sh
   ```

3. Save the changes to the `env.sh` file.

2

Working with TimesTen Scaleout

This chapter focuses on working with TimesTen Grid in SQL Developer.

Topics include:

- Displaying the TimesTen Grid view
- Testing passwordless SSH configuration
- Working with the grid
- Working with the grid model
- Working with physical groups
- Working with hosts
- Working with installations
- Working with management instances
- Working with data instances
- Working with configuration files
- Working with databases
- Working with repositories
- Inspecting logs

Displaying the TimesTen Grid view

SQL Developer enables you to create, manage, and explore a TimesTen Scaleout grid and its components.

To enable the TimesTen Grid view, ensure that you are on the main SQL Developer page. Then, from the **View** menu, select **TimesTen Grid**.

> **Note:** If the TimesTen Grid view option is unavailable, ensure that you have set the environment variables that are described in "Setting environment variables for TimesTen and SQL Developer" on page 1-1.
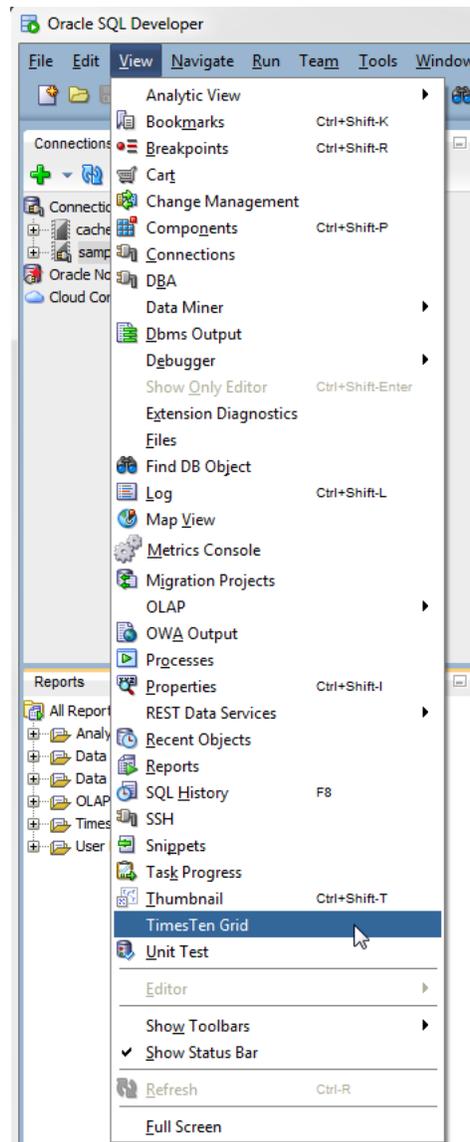
The TimesTen Grid view node displays in the bottom-left corner of the SQL Developer page.

Testing passwordless SSH configuration

Passwordless SSH (secure shell) is required between all intended hosts for a grid in TimesTen Scaleout, regardless of the type of instance (management instance or data instance) on each host. Passwordless SSH is also required for any host that is used as a scp repository. For more information on configuring passwordless SSH and repositories, see "Setting passwordless SSH" and "Working with repositories", respectively, in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

SQL developer provides a tool to test that each host in the grid can access one another through passwordless SSH. To test the passwordless SSH configuration of your hosts, ensure that you are on the main SQL Developer page and that you have enabled the TimesTen Grid view.

1. In the TimesTen Grid view node, right-click the TimesTen Grid folder and select **Test Passwordless SSH**.

Figure 2–2   Select Test Passwordless SSH
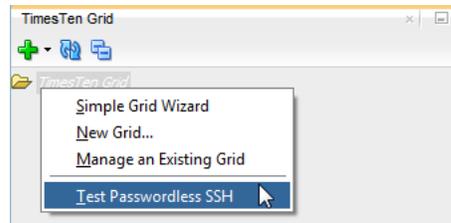
The Test Passwordless SSH configuration dialog displays. Locate the Instance Administrator Credentials region.

2. In the **Instance Administrator** text field, type the username of the instance administrator that will create and manage the grid.

3. By default, the Use Password option is selected. In the **Use Password** text field, type the password of the instance administrator.

Note: Alternatively, you can select the **Use SSH key file** option and use the **Browse** button to specify an SSH key file. The Use SSH key file option enables you to avoid having to specify the password of the instance administrator. You can generate this SSH key file on any of the hosts of your grid as long as you have successfully setup passwordless SSH between these hosts.

Locate the Add Hosts button in the Hosts region.

4. Click **Add Hosts** once for every host that you want to add to the grid topology.

5. In the **Host Address** column of the hosts table, type the host address for every host that will be part of the grid topology.

Note: In case that you are using different internal and external addresses for a host, use the internal host address.

---

6. Click **Test**.

Figure 2–3 Click Test
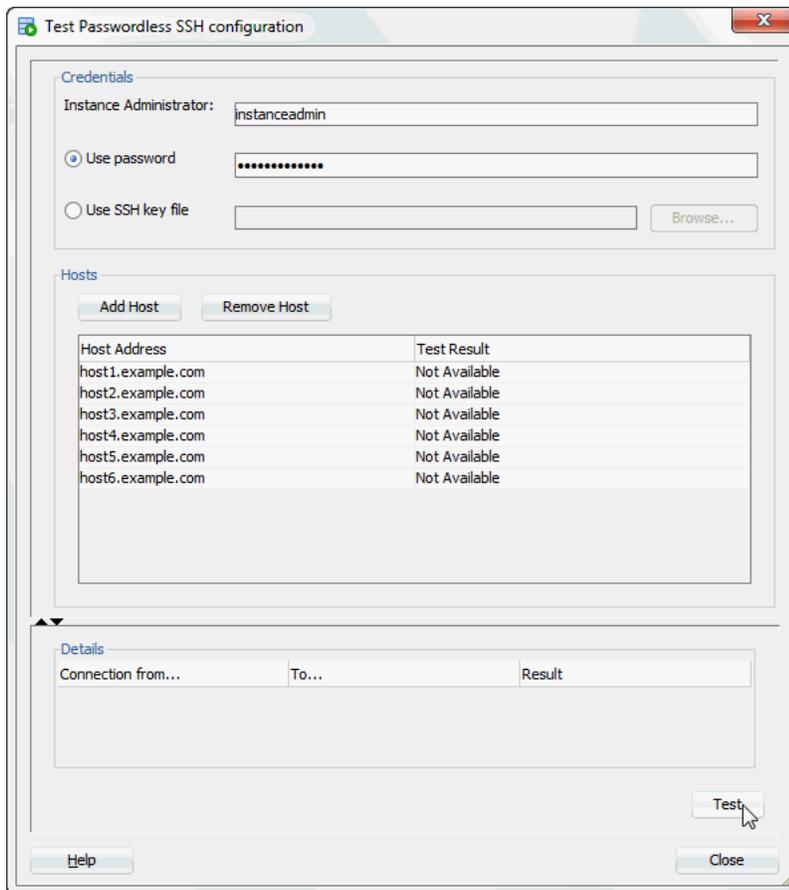
The test can take several minutes depending on the number of hosts that you selected. In the Result column of the Details table and the Test Result column of the Hosts table, you should see the text "OK" for every host. If any host displays the text "Failed", ensure that the particular host is configured for passwordless SSH. For more information on configuring passwordless SSH, see "Setting passwordless SSH" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*. Locate the Close button at the bottom of the dialog.

7. Click **Close**.

The Test Passwordless SSH configuration dialog closes.

You have successfully tested the passwordless SSH configuration between all hosts for a grid in TimesTen Scaleout. You are now ready to create a grid.

Working with the grid

SQL Developer enables you to create a new grid, manage an existing grid, remove a grid from SQL Developer, and perform various grid operations.

- Create a grid

- Manage an existing grid

- Shutdown a grid

- Start a grid

- Remove a grid from SQL Developer

Create a grid

SQL developer enables you to create a grid. You can create a TimesTen Scaleout grid with either the Simple Grid Wizard or the New Grid option. The New Grid option is recommended for advanced users.

Before you create a grid:

- Ensure that you have configured passwordless SSH for all intended hosts for your grid. See "Testing passwordless SSH configuration" on page 2-2 for more information.

- Ensure that there is a common instance administrator user ID and numeric UID across all intended hosts for your grid. For more information, see "Understanding the TimesTen users group and the operating system user" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

- Ensure that you have configured Apache ZooKeeper for TimesTen Scaleout membership services. For more information, see "Configuring Apache ZooKeeper as the membership service" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

Consider these things when you decide whether to use the Simple Grid Wizard or the new grid option:

- The Simple Grid Wizard does not allow you to define physical groups, but you are able to assign a host to a specific data space group. If you want to define physical groups, use the new grid option.

- The Simple Grid Wizard automatically applies the grid model changes at the end of the wizard. If you want to verify the model changes before applying the grid model, use the new grid option.

- The Simple Grid Wizard and the new grid option both create a TimesTen installation on every specified host based in the installation location that you specify.

- The Simple Grid Wizard and the new grid option both create data instance locations on every specified host that does not have a data instance.

- The New Grid option only creates a management instance. For production deployments, it is highly recommended to create a second management instance. After TimesTen creates the management instance you need to add data instances to your grid.

There are several restrictions that you should consider when deciding on the object names for TimesTen Scaleout. For more information, see "Grid objects and object naming" in the *Oracle TimesTen In-Memory Database Reference*.

Follow one of these procedures to create a grid with SQL Developer:

- Create a grid with the Simple Grid Wizard

- Create a grid with the New Grid option

Create a grid with the Simple Grid Wizard

To use the Simple Grid Wizard to create a grid, ensure that you are on the main SQL Developer page and that you have enabled the TimesTen Grid view. To start the Simple Grid Wizard:

In the TimesTen Grid view node, right-click the TimesTen Grid Folder and select **Simple Grid Wizard**.

Figure 2–4   Select Simple Grid Wizard
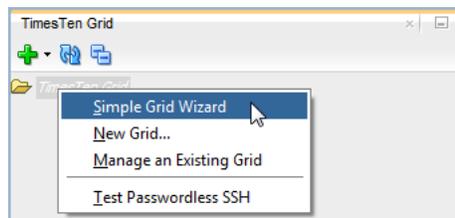
If you are using the Simple Grid Wizard for the first time, a Passwordless SSH dialog displays. Ensure that you have completed the prerequisites that the dialog specifies. If you do not want to see the Passwordless SSH dialog when you use the Simple Grid Wizard, select the **Skip This Message Next Time** check box. Then, click the **Yes** button. See "Create a grid" on page 2-5 for more information on the prerequisites for creating a grid.

You are now ready to configure your initial management instance.

Configure your first management instance To configure the first management instance for your grid:

> **Note:**   The Simple Grid Wizard enables you to a create a single management instance. It is recommended that you create an additional management instance by creating a second management instance. See "Create a new management instance" on page 2-56 for more information.

1. In the **Name** text field, type a name for your grid. Select a name that is a meaningful identifier. The default value is `grid1`.

2. Select a K-safety level from the **K-safety** list. It is recommended that you select a K-safety level of `2`.

3. In the **Membership Configuration** field, specify a ZooKeeper membership configuration file. You can specify the path of the file in the text field or you can use the **Browse** button to locate a file.

   For more information about the ZooKeeper membership configuration file, see "Configuring Apache ZooKeeper as the membership service" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

4. If you want to specify a host name for the management instance that is different from the host name that the operating system assigns, deselect the **Use Default** check box. In the **Host Name** field, type a name for your management instance host. Select a name that is a meaningful identifier.

5. In the **Internal Address** field, specify the internal address of the management instance.

6. In the **External Address** field, specify the external address of the management instance.

7. In the **Instance Name** field, type a name for your management instance. Select a name that is a meaningful identifier. The default value is `instance1`.

8. In the **Instance Location** field, specify the directory where you want TimesTen to create the instance. TimesTen creates a directory with the instance name inside of the directory you specify. For example, if you have an instance name of `instance1` and an instance location of `/TimesTen/instance/`, TimesTen creates your instance in the `/TimesTen/instance/instance1/` directory.

9. In the **Daemon Port** field, specify the port number for the TimesTen daemon. The default value is `6624`.

10. In the **Client/Server Port** field, specify the port number for the TimesTen client/server. The default value is `6625`.

11. In the **Management Port** field, specify the management port number. The default value is `3754`.

   The management port is the value for the TCP/IP port number of the first management instance replication agent.

12. In the **Software Location** field, specify the location of the TimesTen installation file. You can specify the path in the text field or you can use the **Browse** button to locate the file.

13. In the **Installation Location** field, specify the directory where you want TimesTen to copy the install file. TimesTen copies the install file and then extracts the installation files. The installation files are extracted to a directory that is named based on the release number of TimesTen Scaleout. For example, if you are installing TimesTen Scaleout 18.1, the files are extracted to a `tt18.1.1.1.0` directory inside of the installation location that you specified.

14. If you want to specify a user group that will own the TimesTen installation that is different from the primary user group of the instance administrator, deselect the **Use Default** check box. In the **TimesTen Group** field, specify the user group that will own the TimesTen installation. All the installations and instances in the grid will be owned by this user group.

15. In the **Instance Administrator** field, type the instance administrator of the management instance. Ensure that the instance administrator exists on all intended hosts with a common user ID and numeric UID.

16. By default, the Use Password option is selected. In the **Use Password** text field, type the password for the instance administrator.

   > **Note:** Alternatively, you can select the **Use SSH key file** option and use the **Browse** button to specify an SSH key file. The Use SSH key file option enables you to avoid having to specify the password of the instance administrator. You can generate this SSH key file with the ssh-keygen Linux tool on any of the hosts of your grid as long as you have successfully setup passwordless SSH between these hosts.

17. To avoid SQL Developer prompts for the instance administrator password during future operations, select the **Save Password** check box.

   You are now ready to have SQL Developer validate your input values. Locate the Next button.

18. Click **Next**.

Figure 2–5   Click Next
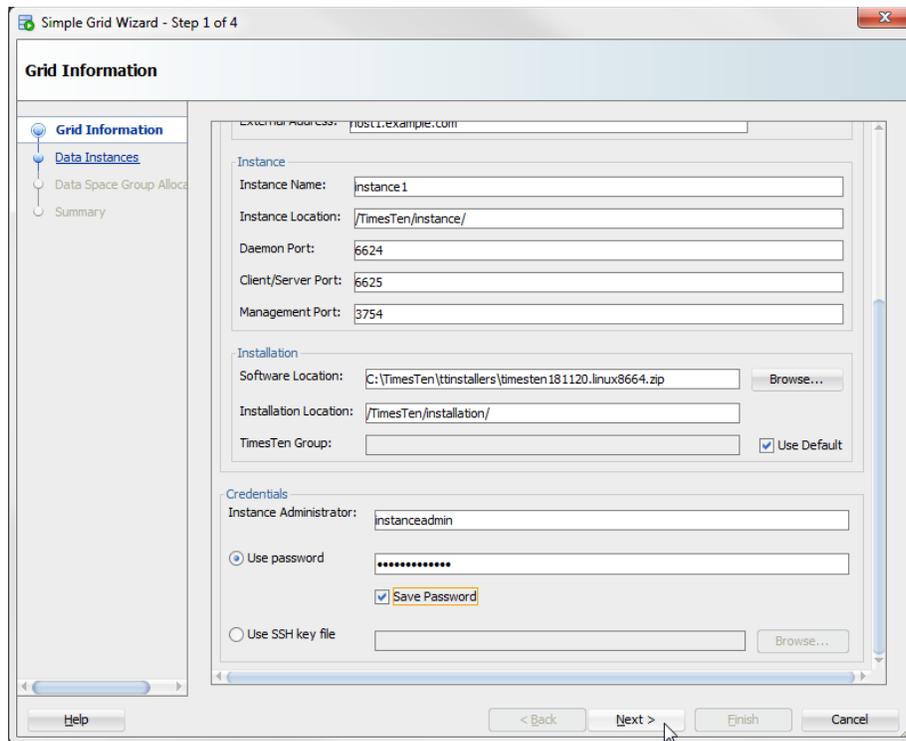
SQL Developer validates your input values and displays a progress dialog. The validation can take a few minutes. If SQL Developer displays an error message, troubleshoot the error and click the Next button again to revalidate your input values.

The progress dialog closes and the Simple Grid Wizard dialog refreshes. You are now ready to add data instances.

Add data instances To add data instances to your grid:

> **Note:**   When you add a data instance, SQL Developer actually adds a host, an installation, and then an instance.

1.   Right-click the Data Instances folder and select **New**.

Figure 2–6   Select New
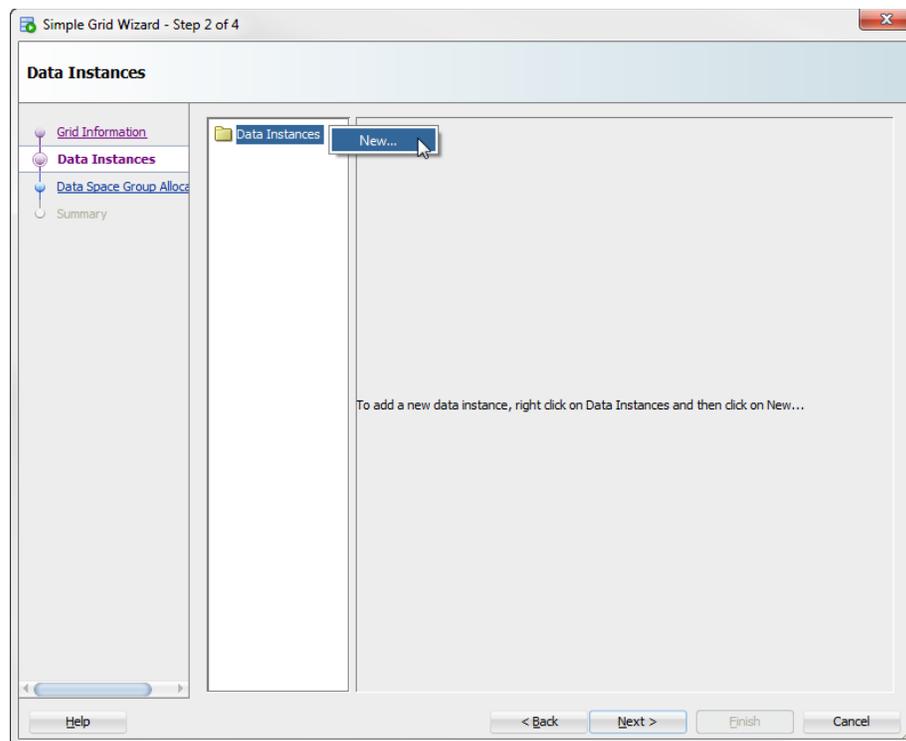
You are now ready to add the information for your data instance. Locate the Host Name text field.

2. If you want to specify a host name for the data instance that is different from the host name that the operating system assigns, deselect the **Use Default** check box. In the **Host Name** field, type a name for your data instance host. Select a name that is a meaningful identifier.

3. In the **Internal Address** field, specify the internal address of the data instance.

4. In the **External Address** field, specify the external address of the data instance.

5. In the **Instance Name** field, type a name for your management instance. Select a name that is a meaningful identifier. The default value is `instance1`.

6. In the **Instance Location** field, specify the directory where you want TimesTen to create the instance.

7. In the **Daemon Port** field, specify the port number for the TimesTen daemon. The default value is `6624`.

8. In the **Client/Server Port** field, specify the port number for the TimesTen client/server. The default value is `6625`.

9. In the **Installation Location** field, specify the directory where you want TimesTen to copy the install file. TimesTen copies the install file and then extracts the installation files. The installation files are extracted to a directory that is named based on the release number of TimesTen Scaleout. For example, if you are installing TimesTen Scaleout 18.1, the files are extracted to a `tt18.1.1.1.0` directory inside of the installation location that you specified.

Locate the Save button.

10. Click **Save** to save the configuration of your data instance.

Figure 2–7   Click Save
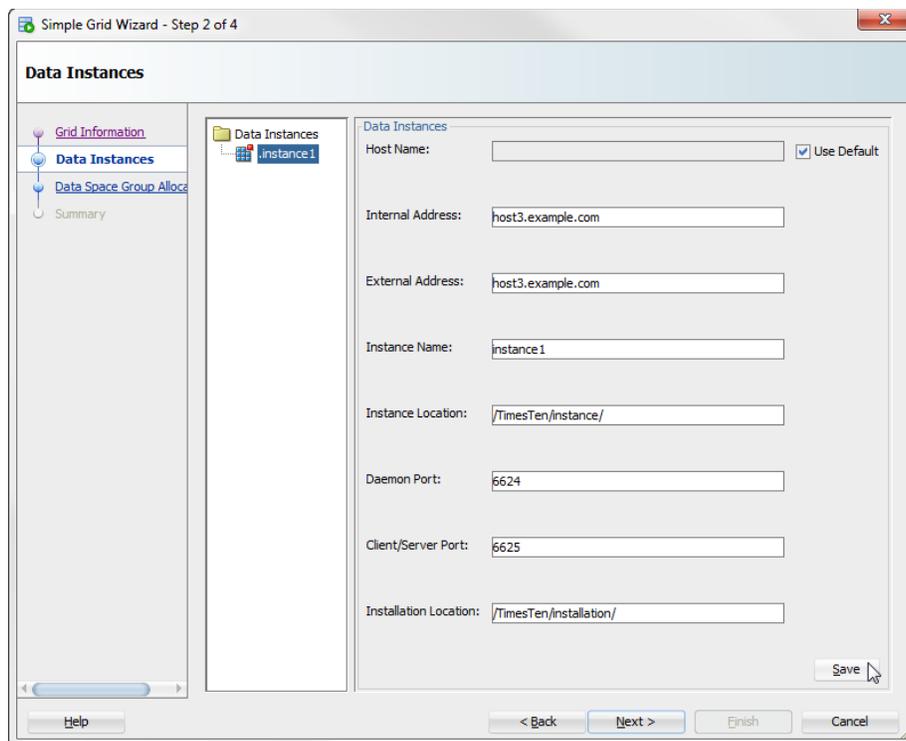
You have successfully added a data instance. When you add a data instance, SQL Developer actually adds a host, an installation, and then an instance.You are now ready to add additional data instances.

You can add additional hosts with installations and data instances by:

- Adding a new data instance.

  Right-click the Data Instances folder and select **New**. Then, follow the steps described in "Add data instances" on page 2-8.

- Duplicating the configuration of another data instance.

  Right-click the data instance that you want to duplicate and select **Duplicate**. SQL Developer adds a new data instance with the same configuration values as the original data instance except for the host name, internal address, and external address. Ensure that you enter the values for the host name, internal address, and external address text fields.

You can remove a data instance by right-clicking the data instance that you want to remove and select **Delete**.

11. Once you have added all of your intended data instances to your grid, click **Next**.

Figure 2–8   Click Next
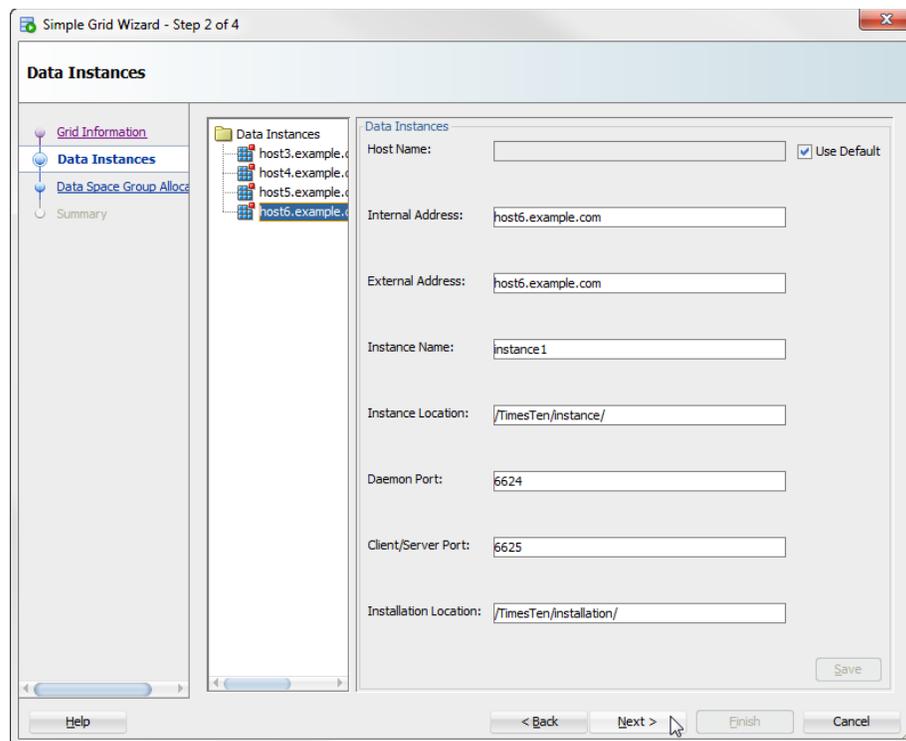
SQL Developer validates your input values and displays a progress dialog. The validation can take a few minutes. If SQL Developer displays an error message, troubleshoot the error and click the Next button again to revalidate your input values.

The progress dialog closes and the Simple Grid Wizard dialog refreshes. You are now ready to assign your hosts to data space groups.

Assign hosts to data space groups Organize the hosts that contain one full copy of the data into a single data space group. Since there are k copies of the data, there are an equal number of data space groups, which are numbered from 1 to k. k is the value that you set for the K-safety configuration of your grid. The hosts in one data space group should be physically separate from the group of hosts in another data space group. Thus, hosts in one data space group are more likely to fail together; hosts in separate data space groups are less likely to fail simultaneously. For more information on data space groups, see "Assigning hosts to data space groups" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

Once you have assigned a host to a data space group and you have applied that change to the grid model, you cannot change the data space group of that host.

To assign hosts to data space groups:

Right-click the **host name** that you want to assign to a data space group. Then, select the data space group where you want to assign the host.

Figure 2–9   Select the data space group
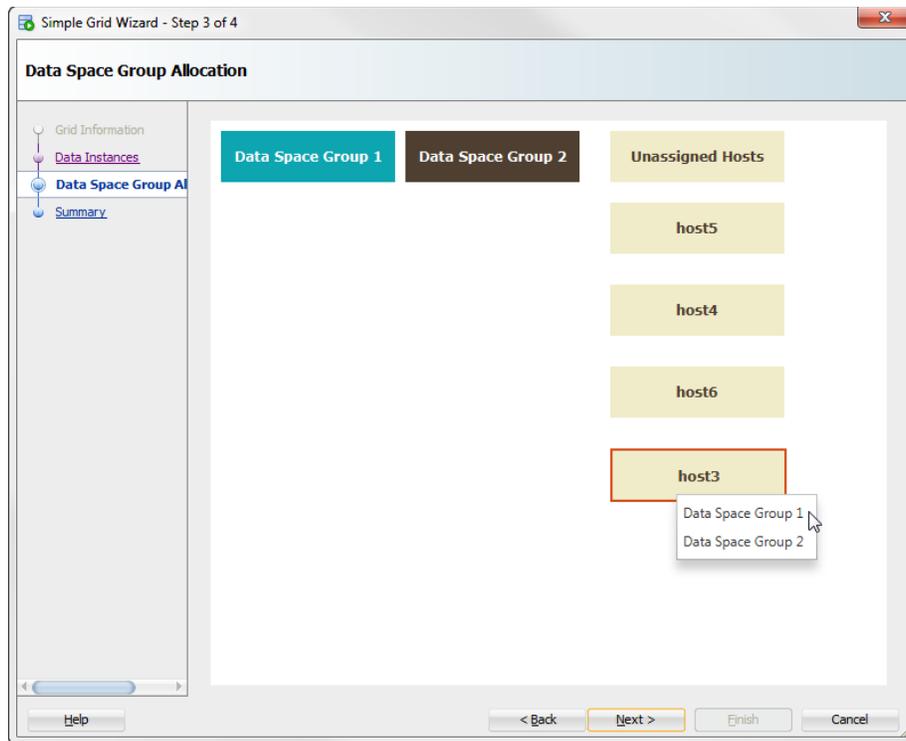
Your host gets assigned to the data space group that you selected. Repeat this procedure to assign all of your hosts to data space groups. Ensure that you have the same amount of hosts in each data space group.

Once you have assigned all of your hosts to data space groups, click **Next**.

Figure 2–10   Click Next
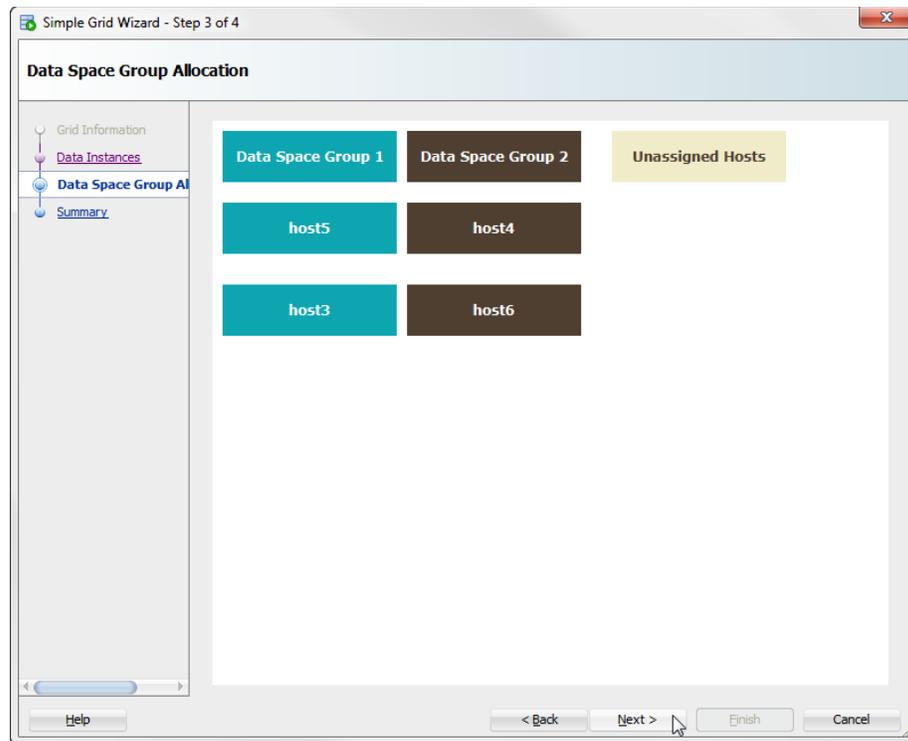
The Simple Grid Wizard refreshes and shows a summary table with the configuration for your grid.

Verify that the configuration settings for your grid are correct and click **Finish**.

Figure 2–11   Click Finish
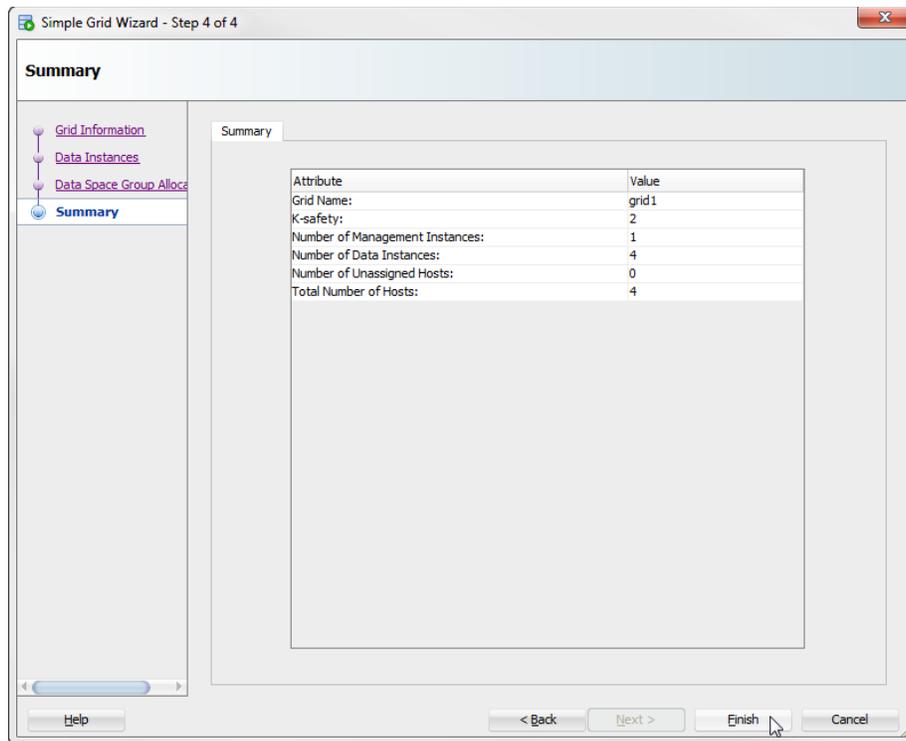
The Simple Grid Wizard dialog closes and a Create Grid progress dialog displays. Creating a grid can take a few minutes depending on the number of hosts. If SQL Developer displays an error message, troubleshoot the error. In the case that an error file is generated, the Create Grid progress dialog displays the error file as a hyperlink. You can save the error file and troubleshoot any errors that are listed in the file.

Create a grid with the New Grid option

To use the new grid option to create a grid, ensure that you are on the main SQL Developer page and that you have enabled the TimesTen Grid view. To create a new grid with the new grid option:

1. In the TimesTen Grid view node, right-click the TimesTen Grid Folder and select **New Grid**.

Figure 2–12   Select New Grid
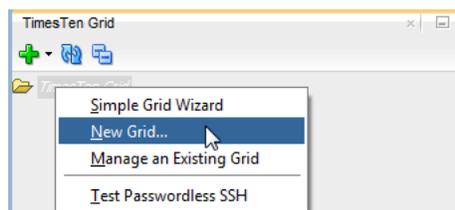
If you are using the new grid option for the first time, a Passwordless SSH dialog displays. Ensure that you have completed the prerequisites that the dialog specifies. If you do not want to see the Passwordless SSH dialog when you use the Simple Grid Wizard, select the **Skip This Message Next Time** check box. Then, click the **Yes** button. See "Create a grid" on page 2-5 for more information on the prerequisites for creating a grid.

You are now ready to configure your initial management instance.

> **Note:** The new grid option enables you to a create a single management instance. See "Create a new management instance" on page 2-56 for more information.

2. In the **Name** text field, type a name for your grid. Select a name that is a meaningful identifier. The default value is `grid1`.

3. Select a K-safety level from the **K-safety** list. It is recommended that you select a K-safety level of `2`.

4. In the **Membership Configuration** field, specify a ZooKeeper membership configuration file. You can specify the path of the file in the text field or you can use the **Browse** button to locate a file.

   For more information about the ZooKeeper membership configuration file, see "Configuring Apache ZooKeeper as the membership service" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

5. If you want to specify a host name for the management instance that is different from the host name that the operating system assigns, deselect the **Use Default** check box. In the **Host Name** field, type a name for your management instance host. Select a name that is a meaningful identifier.

6. In the **Internal Address** field, specify the internal address of the management instance.

7. In the **External Address** field, specify the external address of the management instance.

8. In the **Instance Name** field, type a name for your management instance. Select a name that is a meaningful identifier. The default value is `instance1`.

9. In the **Instance Location** field, specify the directory where you want TimesTen to create the instance. TimesTen creates a directory with the instance name inside of the directory you specify. For example, if you have an instance name of `instance1` and an instance location of `/TimesTen/instance/`, TimesTen creates your instance in the `/TimesTen/instance/instance1/` directory.

10. In the **Daemon Port** field, specify the port number for the TimesTen daemon. The default value is `6624`.

11. In the **Client/Server Port** field, specify the port number for the TimesTen client/server. The default value is `6625`.

12. In the **Management Port** field, specify the management port number. The default value is `3754`.

    The management port is the value for the TCP/IP port number of the first management instance replication agent.

13. In the **Software Location** field, specify the location of the TimesTen installation file. You can specify the path in the text field or you can use the **Browse** button to locate the file.

14. In the **Installation Location** field, specify the directory where you want TimesTen to copy the install file. TimesTen copies the install file and then extracts the installation files. The installation files are extracted to a directory that is named based on the release number of TimesTen Scaleout. For example, if you are

installing TimesTen Scaleout 18.1, the files are extracted to a `tt18.1.1.1.0` directory inside of the installation location that you specified.

15. If you want to specify a user group that will own the TimesTen installation that is different from the primary user group of the instance administrator, deselect the **Use Default** check box. In the **TimesTen Group** field, specify the user group that will own the TimesTen installation. All the installations and instances in the grid will be owned by this user group.

16. In the **Instance Administrator** field, type the instance administrator of the management instance. Ensure that the instance administrator exists on all intended hosts with a common user ID and numeric UID.

17. By default, the Use Password option is selected. In the **Use Password** text field, type the password for the instance administrator.

> **Note:** Alternatively, you can select the **Use SSH key file** option and use the **Browse** button to specify an SSH key file. The Use SSH key file option enables you to avoid having to specify the password of the instance administrator. You can generate this SSH key file on any of the hosts of your grid as long as you have successfully setup passwordless SSH between these hosts.

18. To avoid SQL Developer prompts for the instance administrator password during future operations, select the **Save Password** check box.

You are now ready to have SQL Developer validate your input values. Locate the Create button.

19. Click **Create**.

Figure 2–13   Click Create
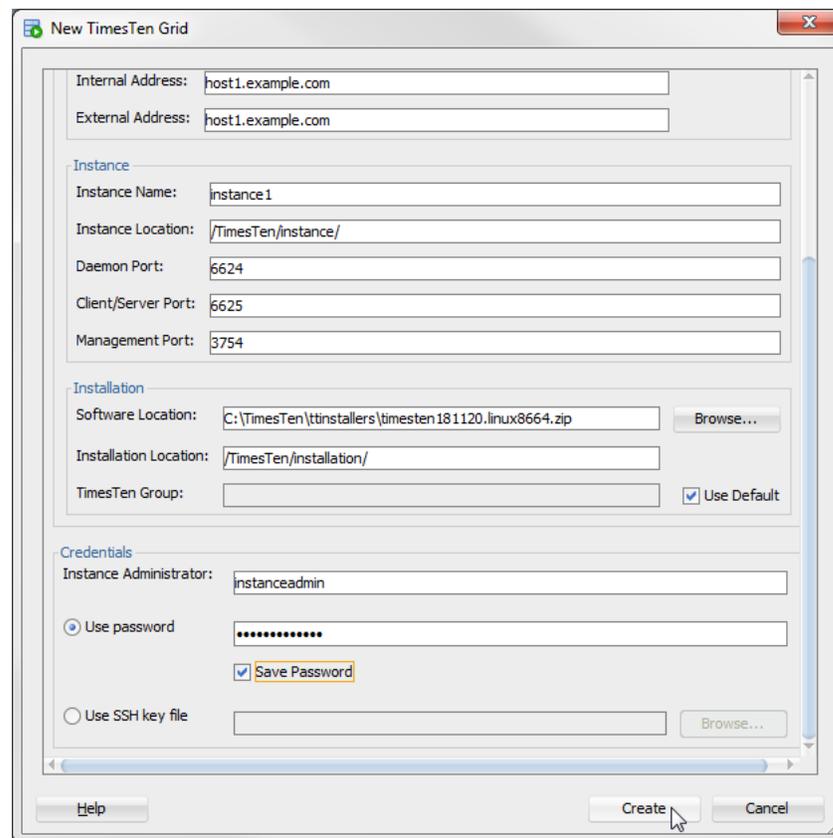
SQL Developer validates your input values and displays a progress dialog. The validation can take a few minutes. If SQL Developer displays an error message, troubleshoot the error and create your grid with the new grid option again.

The New TimesTen Grid dialog closes and the a progress dialog displays. Creating a grid can take a few minutes. If SQL Developer displays an error message, troubleshoot the error. In the case that an error file is generated, the Create Grid progress dialog displays the error file as a hyperlink. You can save the error file and troubleshoot any errors that are listed in the file.

You have successfully created a management instance for your grid. You are now ready to add an additional management host or data instances.

Manage an existing grid

If you created a grid with the `ttGridAdmin` utility, you can add the grid to the TimesTen Grid view. After you add your grid to the TimesTen Grid view you can manage and work with your grid in SQL Developer.

To add an existing grid to the TimesTen Grid view, ensure that you are on the main SQL Developer page and that you have enabled the TimesTen Grid view. To manage an existing grid:

1. In the TimesTen Grid view node, right-click the TimesTen Grid Folder and select **Manage an Existing Grid**.

Figure 2–14   Select Manage an Existing Grid
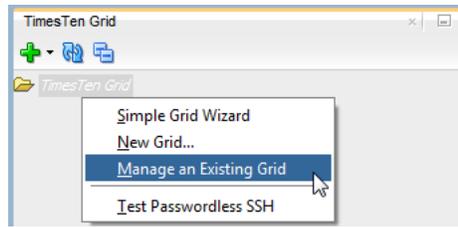
The Manage an Existing Grid dialog displays. Locate the Host Address text field.

2. In the **Host Address** field, specify the external address of the active management instance of your grid.

3. In the **Instance Home** field, specify the instance home directory of your active management instance.

4. In the **Instance Administrator** field, type the instance administrator of the active management instance.

5. By default, the Use Password option is selected. In the **Use Password** text field, type the password for the instance administrator.

Note: Alternatively, you can select the **Use SSH key file** option and use the **Browse** button to specify an SSH key file. The Use SSH key file option enables you to avoid having to specify the password of the instance administrator. You can generate this SSH key file on any of the hosts of your grid as long as you have successfully setup passwordless SSH between these hosts.

---

6. To avoid SQL Developer prompts for the instance administrator password during future operations, select the **Save Password** check box.

    You are now ready to have SQL Developer validate your input values. Locate the Test button.

7. Click **Test**.

Figure 2–15   Click Test
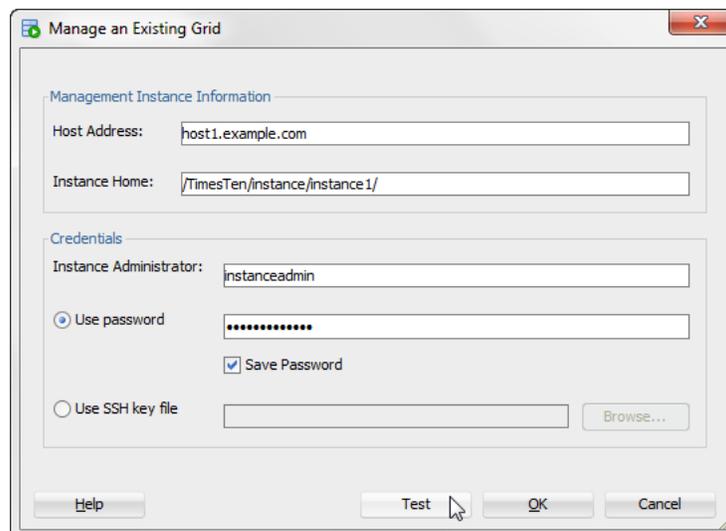
A progress dialog displays. The test process can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

Once the test process completes, a Test Grid Existence dialog displays. Locate the OK button.

8. Click **OK**.

   The Test Grid Existence dialog closes. You are now ready to save the configuration of your existing grid.

   Locate the OK button.

9. Click **OK**.

   SQL Developer validates your input values and displays a progress dialog. The validation can take a few minutes. If SQL Developer displays an error message, troubleshoot the error and fix any invalid input values.

   Once SQL Developer completes validation, a Manage an Existing Grid confirmation dialog displays. Locate the OK button.

10. Click **OK**.

    The Manage an Existing Grid dialog closes.

You have successfully added an existing Grid to SQL Developer. You are now ready to manage and work with your grid in SQL Developer.

Shutdown a grid

SQL Developer enables you to shutdown the components of a grid, such as databases, data instances, and management instances.

To shutdown a grid, ensure that you are on the main SQL Developer page and that you have enabled the TimesTen Grid view.

> **Note:** If you unload a database, stop a data instance, or stop a management instance and the progress dialog does not progress, you can cancel out of the progress dialog. Then, click Next and SQL Developer displays a warning message that a database is loaded or an instance is started. Click Yes to proceed to the next shutdown step.

1. Right-click the **name of your grid**, select **Grid Control**, then select **Shutdown**.

Figure 2–16 Select Shutdown
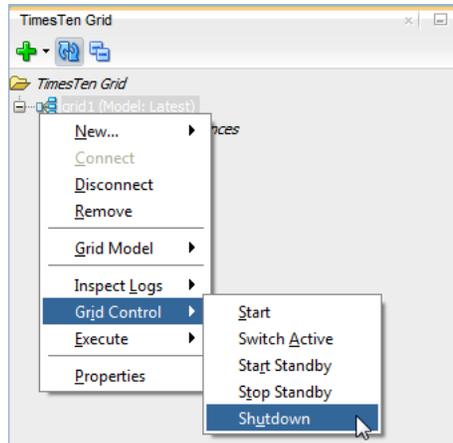
The Shutdown Grid Wizard dialog and Updating databases information progress dialog display.

The Updating databases information progress dialog closes. If a database has open connections, click the link in the Applications Connection column of the database table to view active connections to a specific database. Ensure that you close all applications that are connected to the database before shutting down a grid.

Locate the Unload databases button.

2. Click **Unload databases**.

Figure 2–17   Click Unload databases
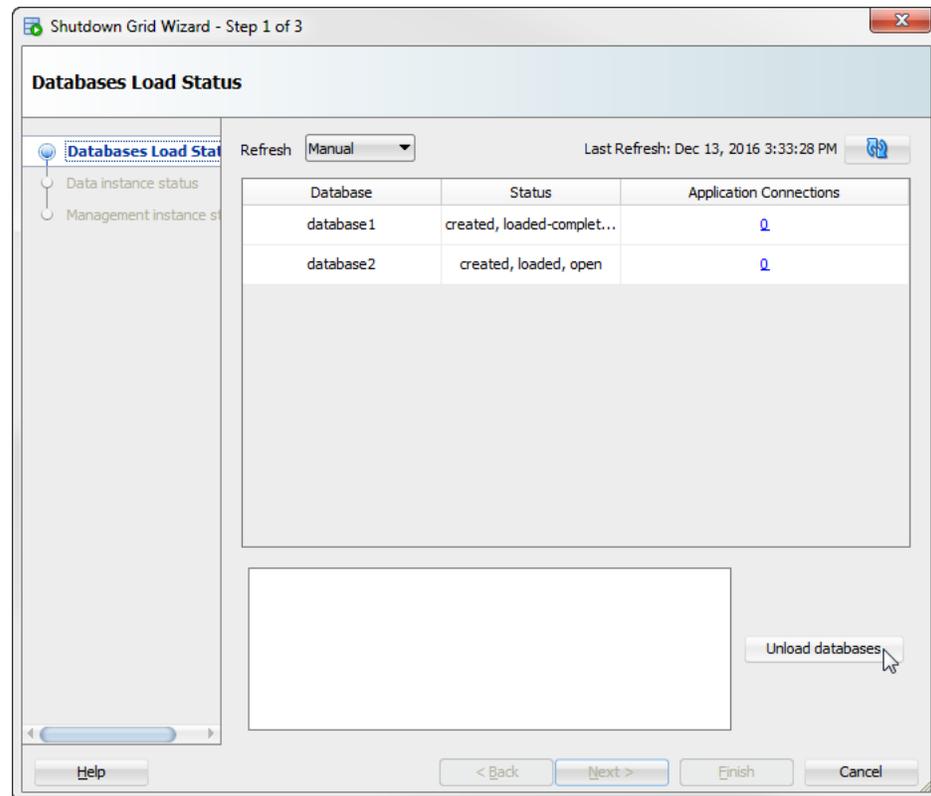
The Updating databases information progress dialog displays. Unloading databases can take a few minutes depending on the number of databases. If SQL Developer displays an error message, troubleshoot the error.

> **Note:**   If you are unable to disconnect an application from a database, SQL Developer may be unable to unload that database. If that is the case, you can cancel the unload databases process and go to step 3 to attempt to unload your data instances.

The Updating databases information progress dialog closes. In the output text area, you should see the text "All databases are unloaded."

Locate the Next button.

3. Click **Next**.

Figure 2–18   Click Next
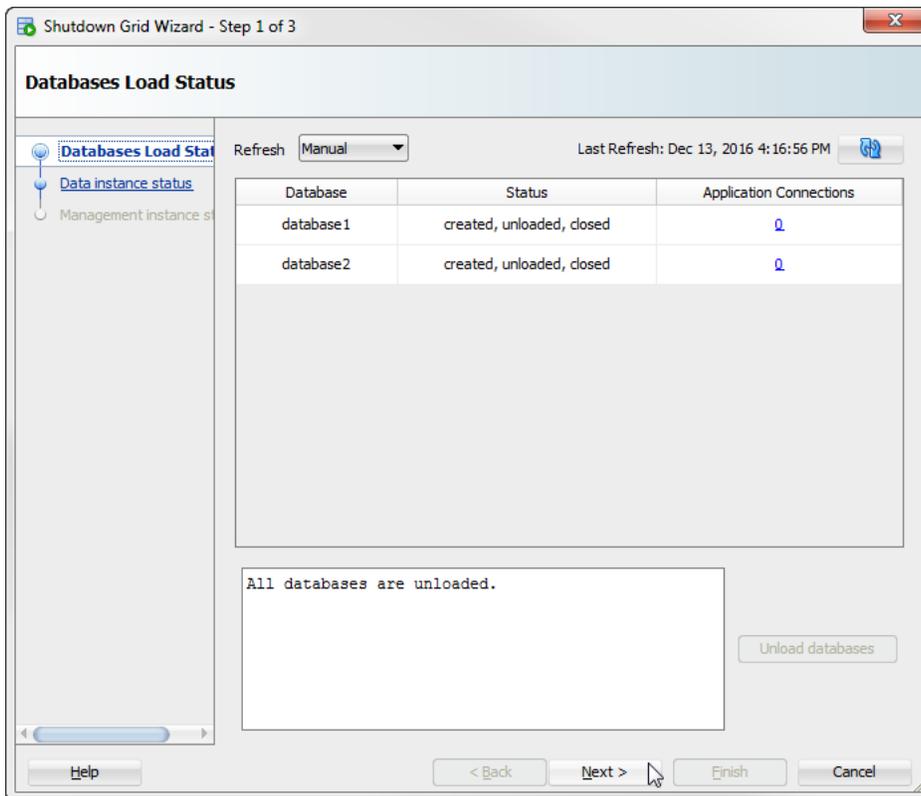
The Shutdown Grid wizard dialog refreshes and the Updating instances information progress dialog displays. The Updating instances information progress dialog closes.

Locate the Stop instances button.

4. Click **Stop Instances**.

> **Note:**   Optionally, select the **Force action** check box of a specific instance to forcefully stop that instance. Forcefully stopping an instance may result in data loss. It is recommended to have a backup of your data before forcefully stopping an instance.

Figure 2–19   Click Stop instances
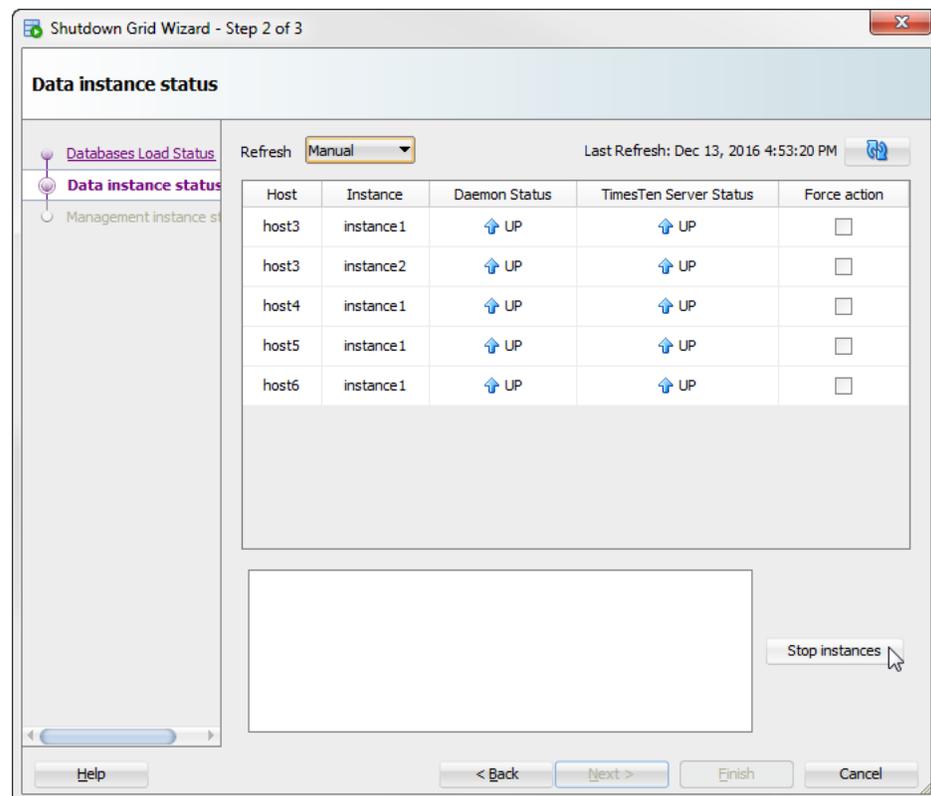
The Updating instances information progress dialog displays. Stopping instances can take a few minutes depending on the number of instances. If SQL Developer displays an error message, troubleshoot the error.

The Updating instances information progress dialog closes. In the output text area, you should see the text "All data instances are down."

Locate the Next button.

5. Click **Next**.

Figure 2–20 Click Next
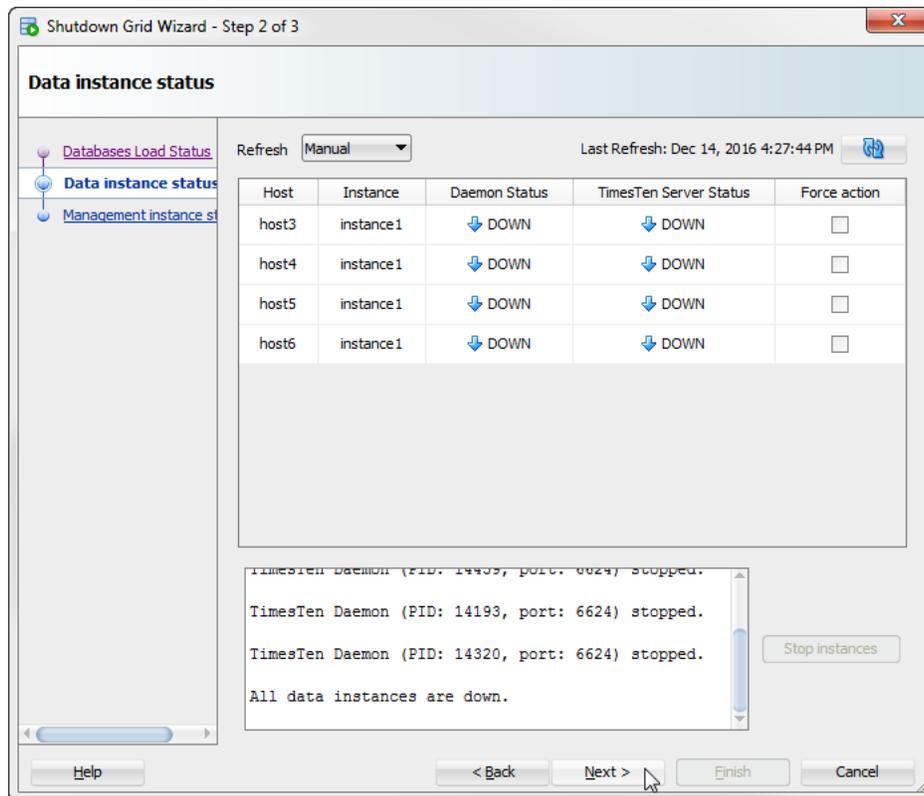
The Shutdown Grid wizard dialog refreshes and the Updating instances information progress dialog displays. The Updating instances information progress dialog closes.

Locate the Stop instances button.

6. Click **Stop instances**.

Figure 2–21   Click Stop instances
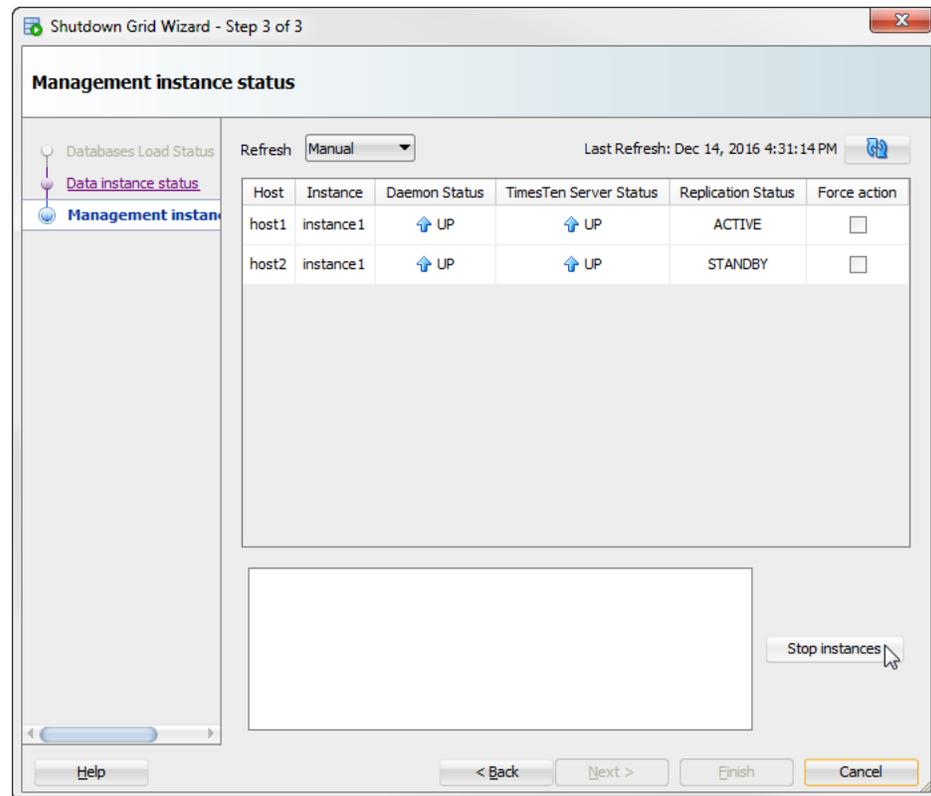
The Updating instances information progress dialog displays. If SQL Developer displays an error message, troubleshoot the error.

The Updating instances information progress dialog closes. In the output text area, you should see the text "All management instances are down."

Locate the Finish button.

7. Click **Finish**.

Figure 2–22   Click Finish
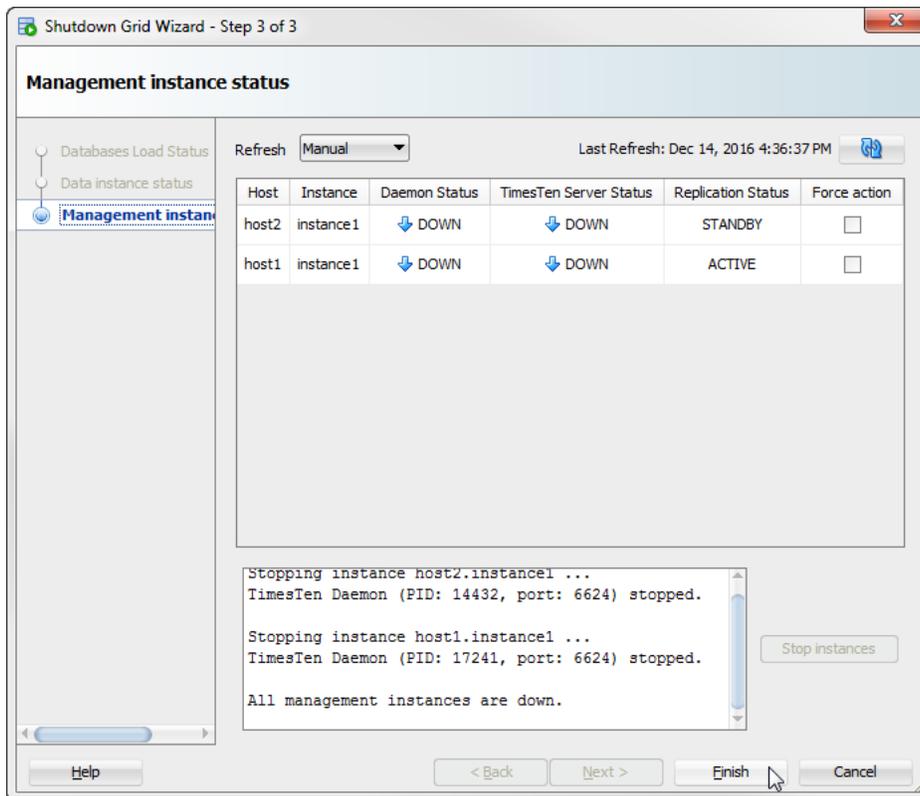
The Shutdown Grid Wizard dialog closes.

You have successfully shutdown the components of a grid, such as databases, data instances, and management instances.

Start a grid

SQL Developer enables you to start the components of a grid, such as management instances, data instances, and databases.

To start a grid, ensure that you are on the main SQL Developer page and that you have enabled the TimesTen Grid view.

> **Note:**   If you load a database, start a data instance, or start a management instance and the progress dialog does not progress, you can cancel out of the progress dialog. Then, click Next and SQL Developer displays a warning message that a database is not loaded or an instance is stopped. Click Yes to proceed to the next startup step.

1. Right-click the **name of your grid**, select **Grid Control**, then select **Start**.

Figure 2–23   Select Start
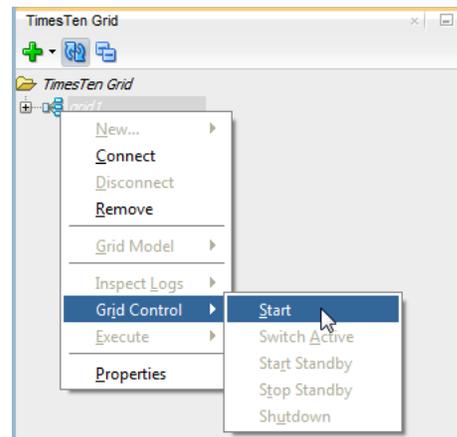
The Start Grid Wizard dialog and Updating instances information progress dialog display. The Updating instances information progress dialog closes.

Locate the Start instances button.

2. Click **Start instances**.

Figure 2–24   Click Start instances
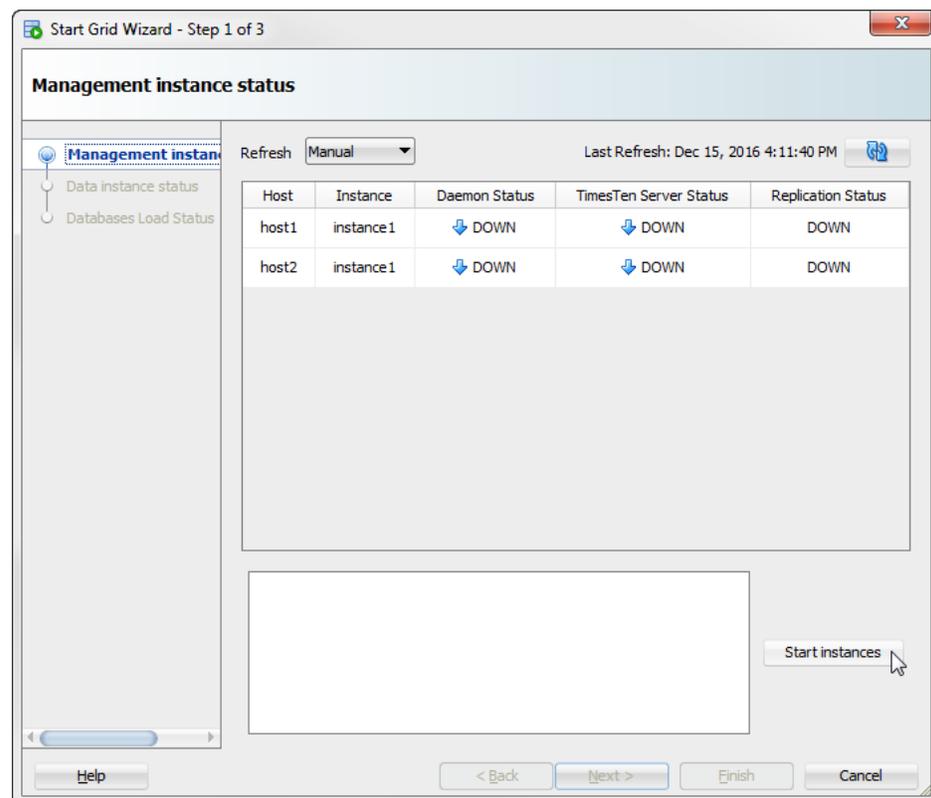
The Updating instances information progress dialog displays. Starting instances can take a few minutes depending on the number of instances. If SQL Developer displays an error message, troubleshoot the error.

The Updating instances information progress dialog closes. In the output text area, you should see the text "All management instances are up and running."

Locate the Next button.

3. Click **Next**.

Figure 2–25   Click Next
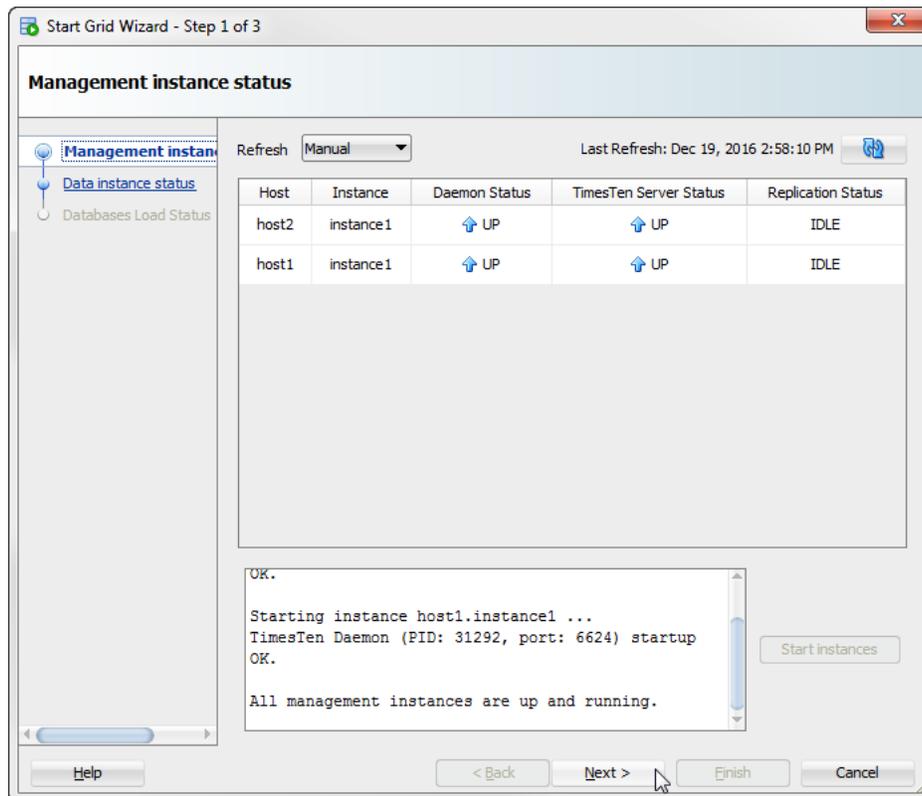
The Start Grid Wizard dialog refreshes and the Updating instances information progress dialog displays. The Updating instances information progress dialog closes.

Locate the Start instances button.

4. Click **Start instances**.

Figure 2–26   Click Start instances
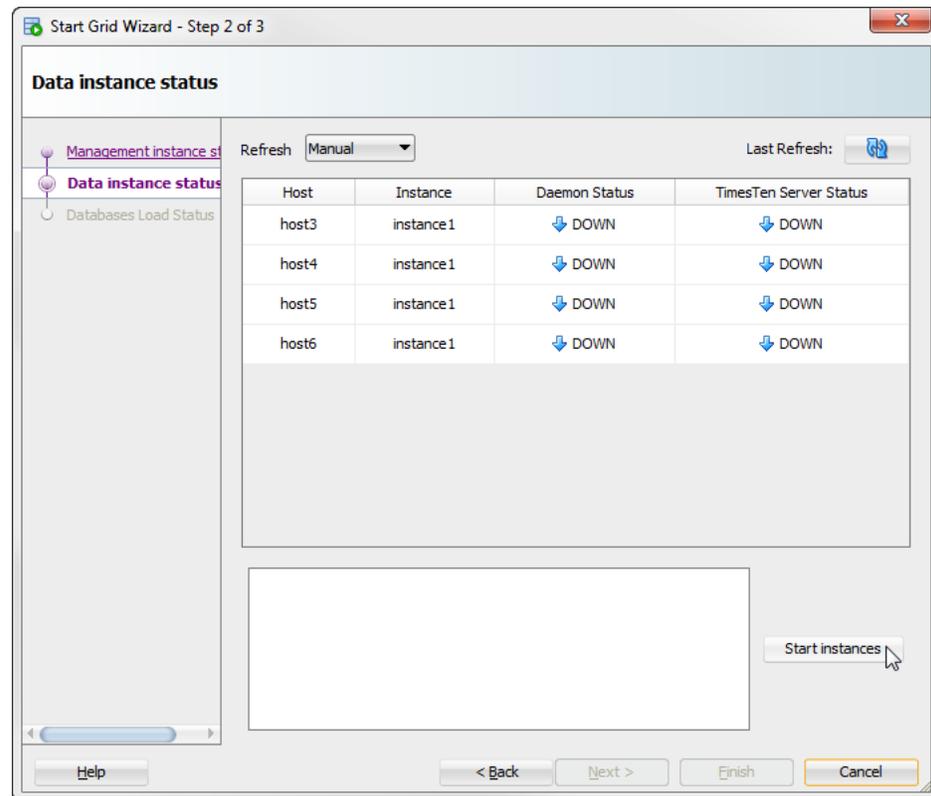
The Updating instances information progress dialog displays. Starting instances can take a few minutes depending on the number of instances. If SQL Developer displays an error message, troubleshoot the error.

The Updating instances information progress dialog closes. In the output text area, you should see the text "All data instances are up and running."

Locate the Next button.

5. Click **Next**.

Figure 2–27   Click Next
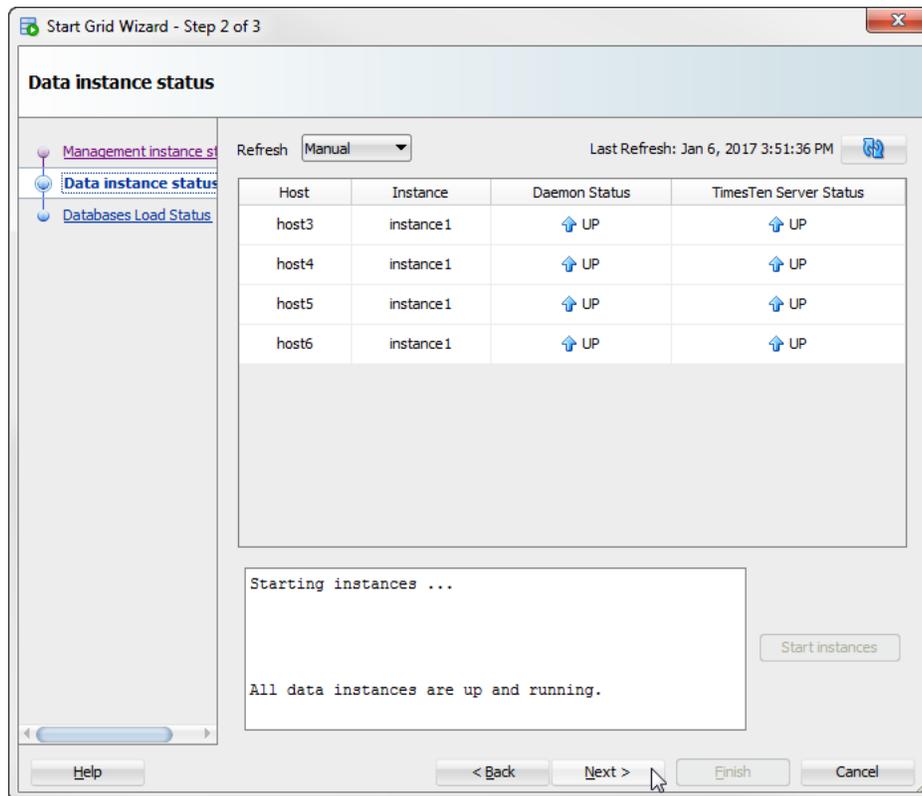
The Updating databases information progress dialog displays. The Updating databases information progress dialog closes.

The Start Grid wizard dialog refreshes.

Locate the Load databases button.

6.   Click **Load databases**.

Figure 2–28   Click Load databases
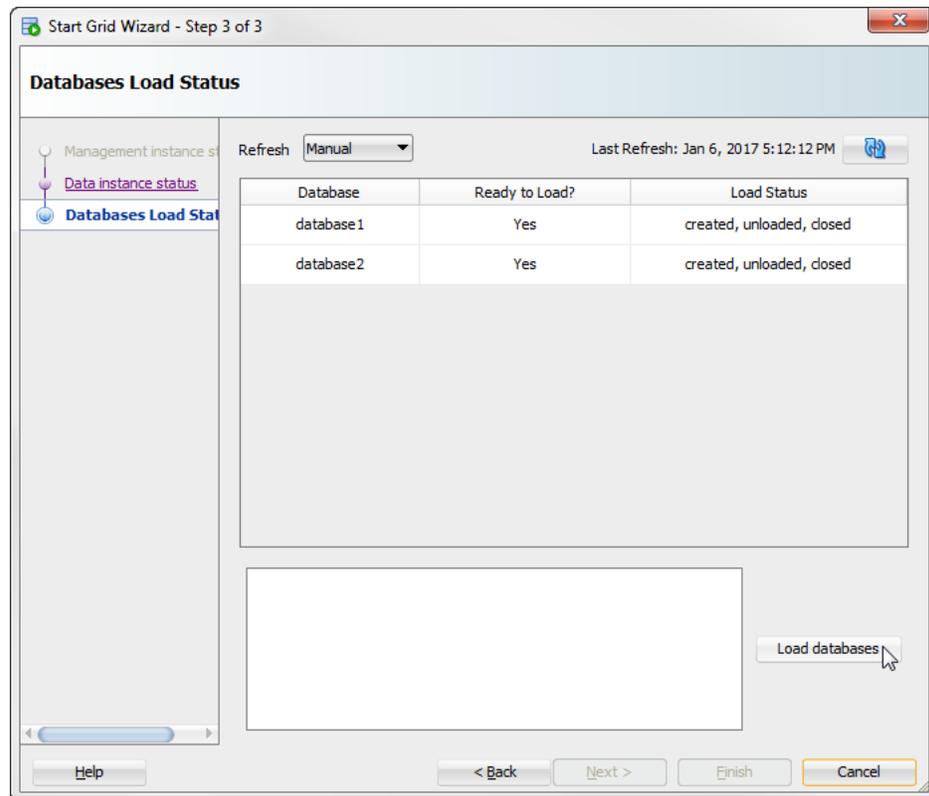
The Updating databases information progress dialog displays. Loading databases can take a few minutes depending on the number of databases. If SQL Developer displays an error message, troubleshoot the error.

The Updating databases information progress dialog closes. In the output text area, you should see the text "All databases are loaded"

Locate the Finish button.

7. Click **Finish**.

Figure 2–29   Click Finish
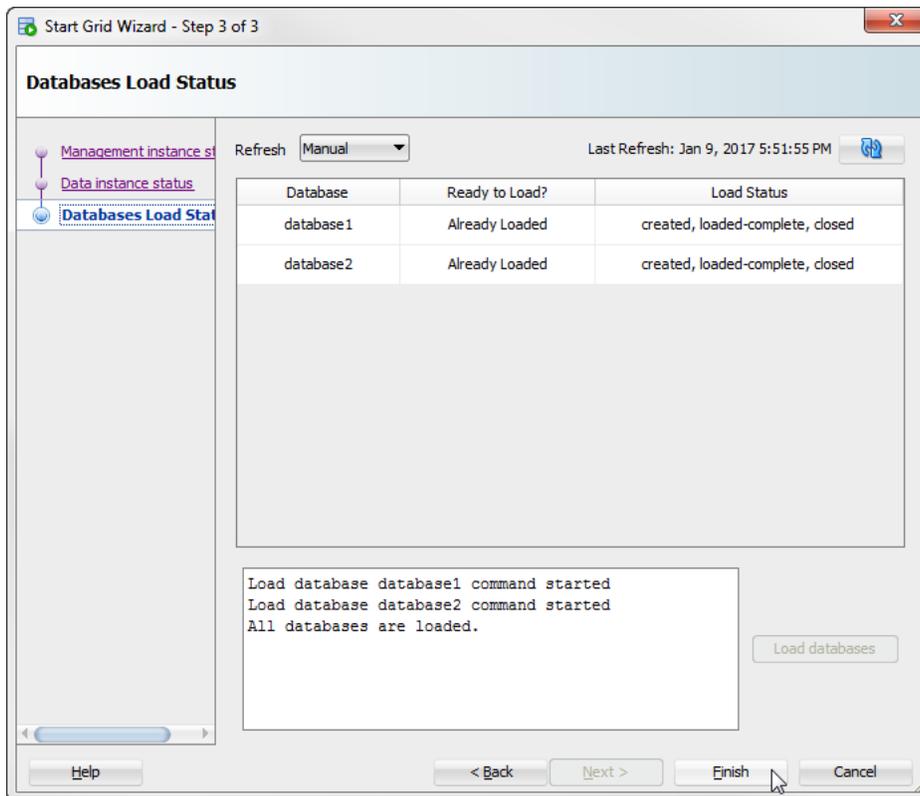
The Start Grid Wizard dialog closes.

You have successfully started your grid. Ensure that you open your database before you attempt to connect applications to the database. See "Perform database operations" on page 2-88 for more information.

Remove a grid from SQL Developer

SQL Developer enables you to remove a grid from SQL Developer. This does not delete a grid and its objects from the hosts on which you installed the grid, but only removes the grid entry from the TimesTen grid view.

To remove a grid, ensure that you are on the main SQL Developer page and that you have enabled the TimesTen Grid view.

Right-click the **name of your grid**, then select **Remove**.

Figure 2–30   Select Remove
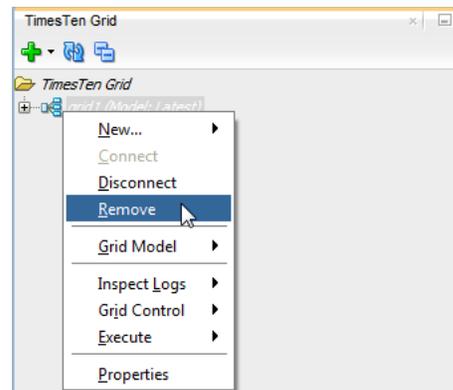
The Remove Grid dialog displays. Locate the Yes button.

Click **Yes**.

The Remove Grid dialog closes and SQL Developer removes the grid entry from the TimesTen grid view.

You have successfully removed a grid entry from the TimesTen grid view.

Working with the grid model

TimesTen Scaleout maintains a single central configuration of the grid within a model that describes the desired structure of a grid, not its current structure. The model represents the desired logical topology of a grid with model objects that is a complete list of all components of a grid, such as installations, hosts, databases, and instances.

SQL Developer enables you to work with a grid model. SQL Developer enables you to perform the following procedures with the grid model:

- Apply grid model changes to the grid

- Export the grid model

- Change the grid model that the grid view displays

- View the changes to the grid model

Apply grid model changes to the grid

After you add more objects to a grid model, you need to apply these components to the grid. Changes made to the model do not immediately impact a grid until you explicitly apply these changes. After you apply the changes, TimesTen Scaleout implements the changes made to the latest version of the model into the operational grid. For example, if you add a new installation and data instance to the latest version of the model, applying the changes to the model performs all of the necessary operations to create and initialize both the installation and the data instance in that host.

Applying grid model changes to the grid is a resource intensive operation. If you plan on making multiple changes to the grid model, make all the changes to your grid model before applying the changes to the grid.

To apply grid model changes to the grid, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded:

1. Right-click the **name of your grid**, select **Grid Model**, then select **Apply Changes**.

Figure 2–31 Select Apply Changes
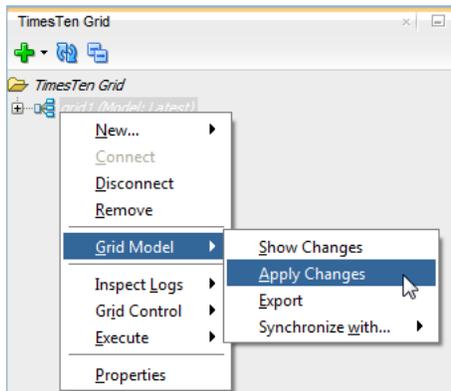
The Comparing current and latest grid dialog model dialog displays. Comparing grid models can take a few minutes.

If SQL Developer does not find any changes between the current and latest grid model, the Comparing current and latest grid dialog model dialog displays the text "Latest model is up to date with current model". Click **Close** to close the dialog.

If SQL Developer finds any changes between the current and latest grid model, a Changes in current grid model dialog displays. Locate the changes table.

2. Review that the changes listed in the table are grid model changes that you want to apply to the grid.

3. Click **Yes**.

Figure 2–32 Click Yes
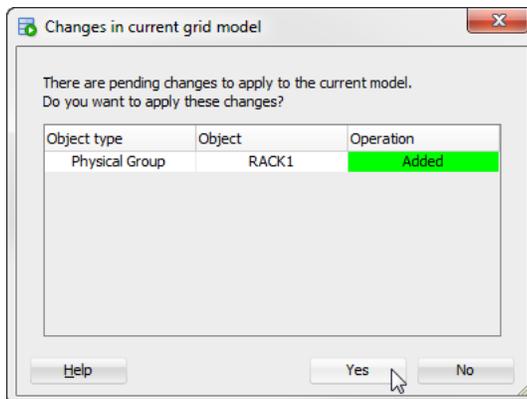
The Changes in current grid model dialog closes and a Apply Changes dialog displays. Applying changes to the grid model can take a few minutes depending on the number of grid model changes. If SQL Developer displays an error message, troubleshoot the error.

4. Click **Close**.

The Apply Changes dialog closes.

You have successfully applied the grid model changes to your grid.

Export the grid model

SQL Developer enables you to export the current model of your grid.

To export the current model of your grid, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Right-click the **name of your grid**, select **Grid Model**, then select **Export**.

Figure 2–33   Select Export
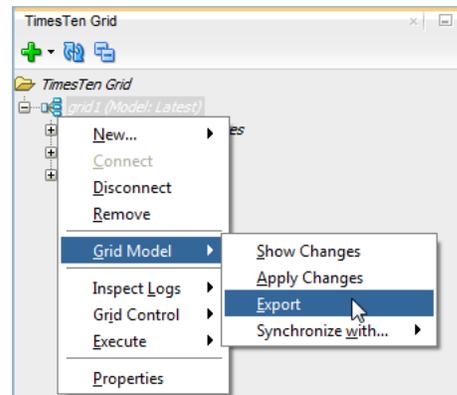
The Export Grid Model dialog displays. Locate the Choose Version to Export region.

2. From the Choose Version to Export region, select **a version of the configuration to export**. These are the options:

    ■ **Current** - This is the current grid model that is applied to the grid.

    ■ **Latest** - This is the latest grid model that is not yet applied to the grid.

    ■ **Version** - This is the version of the configuration file that you can select. If you select this option, use the text field to the right of the Version radio button to select the version to export.

    Locate the Browse button.

3. Click **Browse**.

    The Export file dialog displays. Specify the location of where you want to export the grid model file.

4. In the **File Name** text field, specify a meaningful name for the **grid model file** and click **Save**.

    The Export file dialog closes. In the Export dialog, locate the Export button.

5. Click **Export**.

Figure 2–34   Click Export
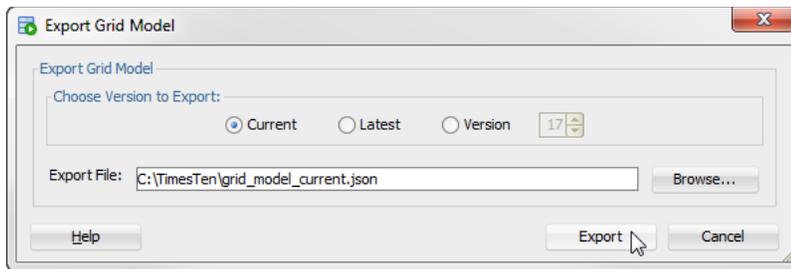
The Verifying File Location progress dialog displays. If SQL Developer displays an error message, troubleshoot the error.

The Verifying File Location progress dialog and the Export Grid Model dialog close. The Export grid model progress dialog displays. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button.

6. Click **Close**.

The Export grid model progress dialog closes.

You have successfully exported a grid model file.

Change the grid model that the grid view displays

SQL Developer enables you to select which grid model shows in the grid view of SQL Developer.

To change the grid model that your grid view displays, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

Right-click your grid node, select **Grid Model**, select **Synchronize with**, then select one of these options:

- **Latest** - Shows all of the latest object changes that have not been applied to the grid model.

- **Current** - Shows only the objects that have been applied to your grid model.

Figure 2–35   Select an option
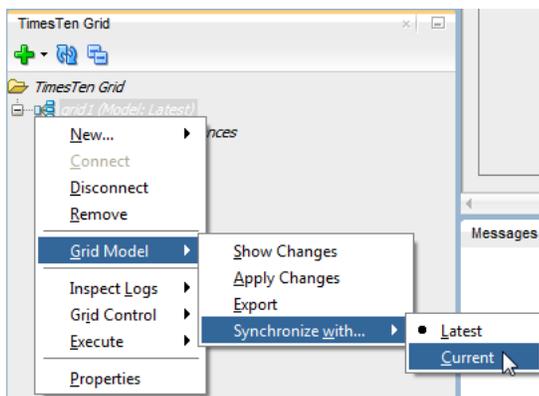
SQL Developer changes the grid model that your grid view displays.

You have successfully changed the grid model that your grid view displays.

View the changes to the grid model

SQL Developer enables you to view the changes between your current and latest grid model. These changes are not yet applied to your grid model.

To view the changes to your grid model, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

Right-click your grid node, select **Grid Model**, then select **Show Changes**.

Figure 2–36   Select Show Changes
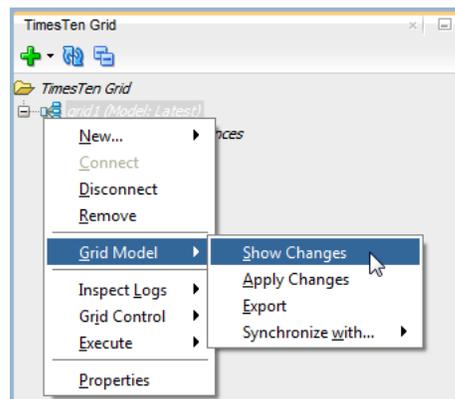
The Comparing current and latest grid model progress dialog displays.

If there are no changes between your current and latest grid model, you see the message "Latest model is up to date with current model". Click **Close**.

If there are changes between your current and latest grid model, the Comparing current and latest grid model progress dialog closes and the Changes in current grid model dialog displays. Review the changes to your grid mode and click **Close**.

You have successfully viewed the changes between your current and latest grid model.

Working with physical groups

SQL Developer enables you to add and remove physical groups from your grid model. You can also assign and remove hosts from physical groups. For more information about physical groups, see "Description of the physical topography of the grid" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

SQL Developer enables you to perform the following procedures with physical groups:

- Define a new physical group
- Assign or remove hosts from a physical group
- Delete a physical group

Define a new physical group

To add a physical group to the grid model, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Expand the **Grid Model** option, right-click the **Physical Groups** option, then select **New Physical Group**.

Figure 2–37   Select New Physical Group
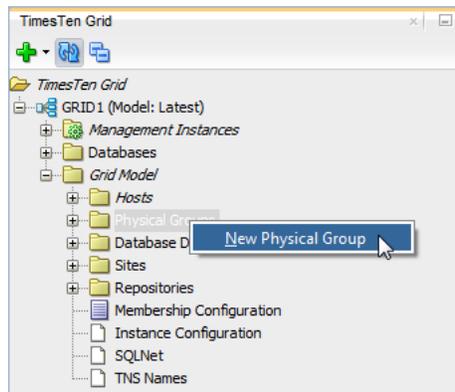
The New Physical Group dialog displays. Locate the Physical Group Name text field. Select a name that is a meaningful identifier.

2. In the **Physical Group name** text field, specify the name of the physical group. Select a name that is a meaningful identifier.

> **Note:**   Step 3 is only required if you want to assign an existing host to the physical group. If you do not want to assign an existing host to the physical group, skip to step 4. Once you add a new host to the grid model, you are able to assign that host to a specific physical group.

3. In the **Excluded** list, locate the row that contains the host name that you want to assign to your new physical group. Double click the host name. If you want to assign multiple hosts, double click the corresponding host name of any additional host.

4. Click **OK**.

Figure 2–38   Click OK
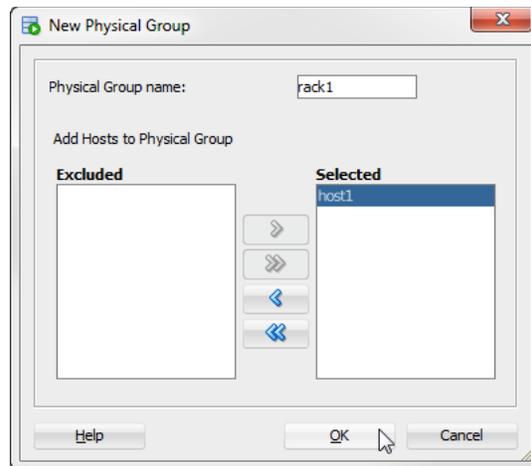
The New Physical Group dialog closes and a Creating Physical Group progress dialog displays. Creating a physical group can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

If you are adding or editing a grid model object for the first time, a Apply Changes dialog displays. This dialog shows you a reminder to apply your changes to the grid model. If you do not want to see this dialog when you add or edit a grid model object, select the **Skip This Message Next Time** check box. Then, click the **OK** button.

Locate the Close button in the Creating Physical Group progress dialog.

5. Click **Close**.

The Creating Physical Group progress dialog closes.

You have successfully added a physical group to the grid model. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes.

Assign or remove hosts from a physical group

To edit a physical group, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Expand the **Grid Model** option, expand the **Physical Groups** option, right-click the **name of the physical group**, then select **Edit**.

Figure 2–39   Select Edit
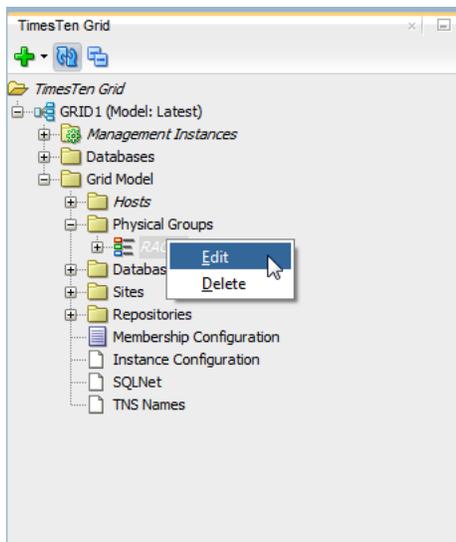
The Add Hosts to Physical Group dialog displays. Locate the Excluded and Selected lists. Identify the hosts that you want to assign or remove from the physical group.

2.   In the **Excluded** list, locate the row that contains the host name that you want to assign to your physical group. Double click the host name. If you want to assign multiple hosts, double click the corresponding host name of any additional host.

3.   In the **Selected** list, locate the row that contains the host name that you want to remove from your physical group. Double click the host name. If you want to remove multiple hosts, double click the corresponding host name of any additional host.

Review the Excluded list to ensure that the list contains the hosts that you want to remove from your physical group. Review the Selected list to ensure that the list contains the hosts that you want to assign to your physical group. Locate the OK button.

4.   Click **OK**.

Figure 2–40   Click OK
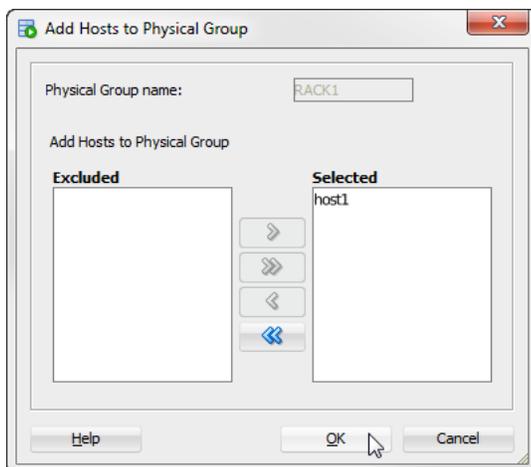
The Add Hosts to Physical Group dialog closes and a Assigning Physical Group progress dialog displays. Assigning hosts to physical groups can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button in the Add Hosts to Physical Group progress dialog.

5. Click **Close**.

You have successfully edited a physical group in the grid model. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

Delete a physical group

To delete a physical group, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Expand the **Grid Model** option, expand the **Physical Groups** option, right-click the **name of the physical group**, then select **Delete**.

Figure 2–41   Select Delete
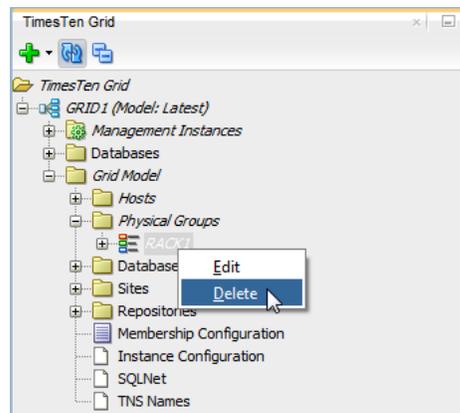
The Delete Physical Group dialog displays. Locate the Yes button.

2. Click **Yes**.

The Delete Physical Group dialog closes and a Delete Physical Group progress dialog displays. Removing a physical group can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button in the Delete Physical Group progress dialog.

3. Click **Close**.

You have successfully removed a physical group from the grid model. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

Working with hosts

SQL Developer enables you to add and remove hosts that you can use for data and management instances. You can also assign hosts to data space and physical groups and execute commands on hosts. SQL Developer enables you to perform the following procedures with hosts:

- Add a host
- Edit a host
- Assign hosts to data space groups
- Delete a host
- Execute a command on a host

Add a host

Before configuring a data or management on a new host you need to add that host to the grid model. It is recommended that you only configure one instance per host. You can either add a host where you define all configuration parameters or by cloning the configuration of an existing host.

- Add a new host
- Clone a host

Add a new host

To add a new host to the grid model, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Expand the **Grid Model** option, right-click the **Hosts** option, then select **New Host**.

Figure 2–42   Select New Host
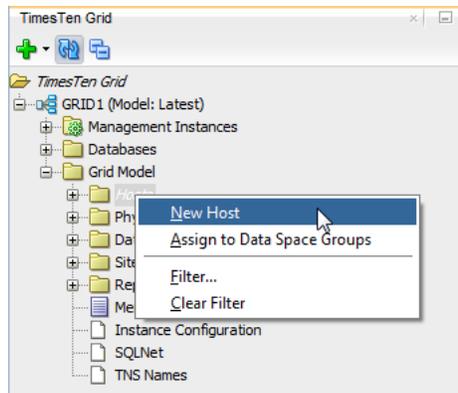
The New Host dialog displays. Locate the Host Name text field.

In some cases, a Passwordless SSH dialog displays. Ensure that you have completed the prerequisites that the dialog specifies. If you do not want to see the Passwordless SSH dialog again, select the **Skip This Message Next Time** check box. Then, click the **Yes** button. See "Create a grid" on page 2-5 for more information on the prerequisites for creating a grid.

2. If you want to specify a host name for the host that is different from the host name that the operating system assigns, deselect the **Use Default** check box. In the **Host Name** field, type a name for your host. Select a name that is a meaningful identifier.

3. In the **Internal Address** field, specify the internal address of the host.

4. In the **External Address** field, specify the external address of the host.

5. From the **Data Space Group** drop-down list, select the data space group where you want to assign the host.

> **Note:** Steps 6-8 are only required if you are using physical groups. If you do not want to use physical groups, skip to step 9. See "Working with physical groups" on page 2-37 for more information on physical groups.

6. To select a physical group for this host, click the **Select** button to the right of the **Physical Group(s)** field.

   The Physical Group(s) dialog displays. Locate the Excluded list.

7. In the **Excluded** list, locate the row that contains the physical group to which you want to assign to your host. Double click the physical group name. If you want to assign your host to multiple physical groups, double click the corresponding physical group name of any additional physical group.

   Review the Selected list to ensure that the list contains the physical groups to which you want to assign your host. Locate the OK button.

8. Click **OK**.

Figure 2–43   Click OK
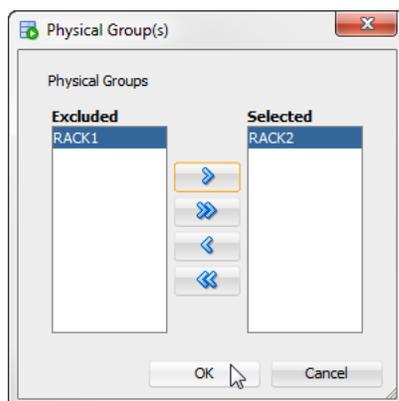
The Physical Group(s) dialog closes and the New Host dialog refreshes. Locate the OK button in the New Host dialog.

9. Click **OK**.

Figure 2–44   Click OK
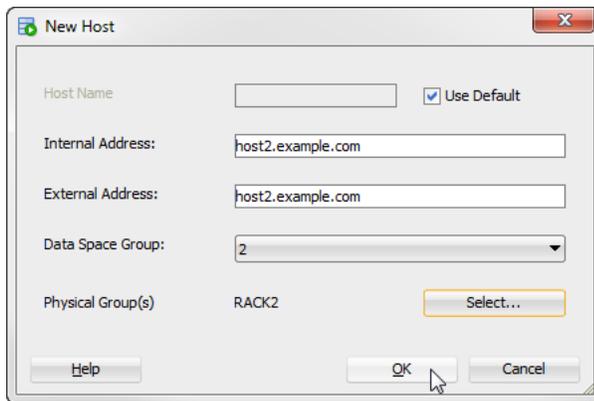
The New Host dialog closes and a New Host progress dialog displays. Adding a new host can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button in the New Host progress dialog.

10. Click **Close**.

You have successfully added a new host to the grid model. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

Clone a host

Cloning a host enables you to create a host for an instance (management or data) by duplicating the configuration of an existing host, including its associated installations and instances.

To clone a host, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Expand the **Grid Model** option, expand the **Hosts** options, right-click the **host that you want to clone**, then select **Clone**.

Figure 2–45   Select Clone
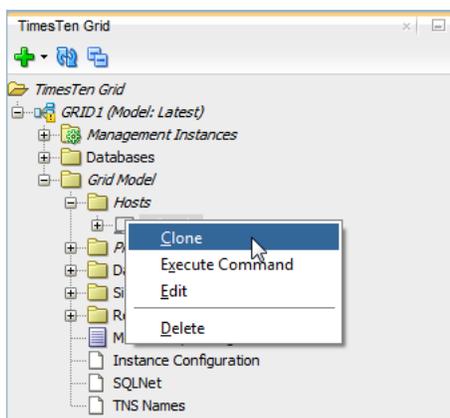
In some cases, a Passwordless SSH dialog displays. Ensure that you have completed the prerequisites that the dialog specifies. If you do not want to see the Passwordless SSH dialog again, select the **Skip This Message Next Time** check box. Then, click the **Yes** button. See "Create a grid" on page 2-5 for more information on the prerequisites for creating a grid.

If you are cloning a management instance for the first time, a Management Instance dialog displays. Ensure that you are sure that you want to clone a management instance. If you do not want to see the Management Instance dialog when you clone a management instance, select the **Skip This Message Next Time** check box. Then, click the **Yes** button.

The Clone Host dialog displays.

2. If you want to specify a host name for the host that is different from the host name that the operating system assigns, deselect the **Use Default** check box. In the **Host Name** field, type a name for your host. Select a name that is a meaningful identifier.

3. In the **Internal Address** field, specify the internal address of the host.

4. In the **External Address** field, specify the external address of the host.

You are now ready to clone the host. Locate the OK button.

5. Click **OK**.

Figure 2–46   Click OK
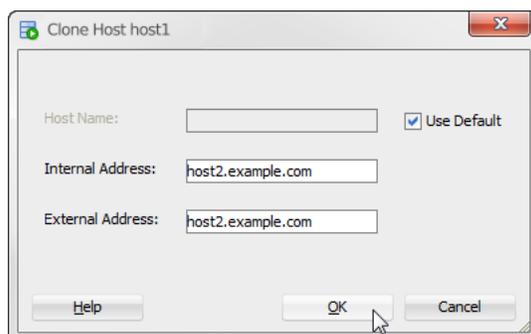
The Clone Host dialog closes and a Clone Host progress dialog displays. Cloning a host can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button in the Clone Host progress dialog.

6. Click **Close**.

Edit the host to change the physical group or data space group of this host. You can only change the data space group of this new host before applying model changes to the grid. See "Edit a host" on page 2-45 for more information.

You have successfully cloned a host. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

Edit a host

SQL Developer enables you to edit the data space group and physical group of a host.

To edit a host, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Expand the **Grid Model** option, expand the **Hosts** options, right-click the **host that you want to edit**, then select **Edit**.

Figure 2–47   Select Edit
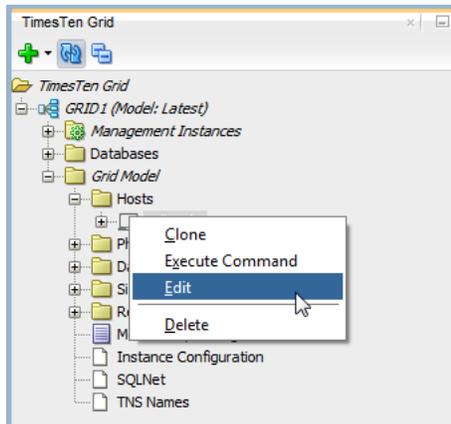
The Edit Host dialog displays.

2. From the **Data Space Group** drop-down list, select the data space group where you want to assign the host.

   Once you have assigned a host to a data space group and you have applied that change to the grid model, you cannot change the data space group of that host.

3. To select a physical group for this host, click the **Select** button to the right of the **Physical Group(s)** field.

   The Physical Group(s) dialog displays. Locate the Excluded and Selected lists. Identify the physical group(s) to which you want to assign your host.

4. In the **Excluded** list, locate the row that contains the physical group to which you want to assign your host. Double click the host name. If you want to assign your host to multiple physical groups, double click the corresponding physical group name of any additional physical group.

5. In the **Selected** list, locate the row that contains the physical group from which you want to withdraw your host. Double click the host name. If you want to withdraw multiple physical groups, double click the corresponding physical group name of any additional physical group.

   Review the Excluded list to ensure that the list contains the physical groups from which you want to withdraw your host. Review the Selected list to ensure that the list contains the physical groups to which you want to assign your host. Locate the OK button.

6. Click **OK**.

   The Physical Group(s) dialog closes. Locate the OK button in the Edit Host dialog.

7. Click **OK**.

   The Edit Host dialog closes and the Edit Host progress dialog displays. Editing a host can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button in the Edit Host progress dialog.

8. Click **Close**.

You have successfully edited a host. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

Assign hosts to data space groups

SQL Developer enables you to assign hosts to data space groups as long as the host is not already associated with a data space group. Once a host is assigned to a data space group, you cannot reassign that host.

To assign hosts to data space groups, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen grid view, and that your grid node is expanded.

1. Expand the **Grid Model** option, right-click the **Hosts** options, then select **Assign to Data Space Groups**.

Figure 2–48   Select Assign to Data Space Groups
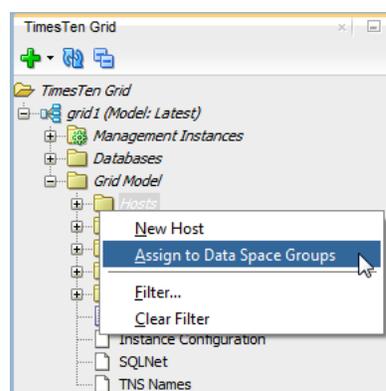
The Assign Hosts to Data Space Groups dialog displays. Locate the host that you want to assign to a data space group.

2. Right-click the **host name** that you want to assign to a data space group. Then, select the data space group where you want to assign the host.

> **Note:** If you have configured physical groups, you can click the **Suggest Assignment** button to have TimesTen Scaleout suggest data space groups for your hosts. Ensure that you have configured physical groups before using the Suggest Assignment button. See "Working with physical groups" on page 2-37 for more information.

Your host gets assigned to the data space group that you selected. Repeat this procedure to assign all of your hosts to data space groups. Ensure that you have the same amount of hosts in each data space group.

3. Once you have assigned all of your hosts to data space groups, click **OK**.

Figure 2–49   Click OK
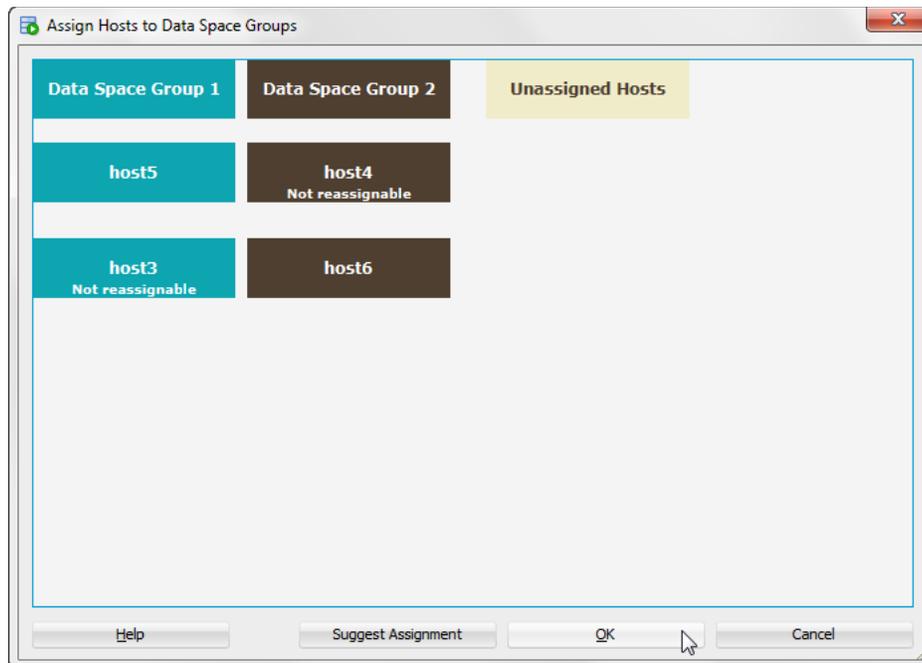
> The Assign Hosts to Data Space Groups progress dialog displays. Assigning hosts to a data space group can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

> The Assign Hosts to Data Space Groups progress dialog and the Assign Hosts to Data Space Groups dialog close.

You have successfully assigned hosts to data space groups. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

Delete a host

SQL Developer enables you to delete a host from your grid model. Before you delete a host, ensure that you have migrated all the data from the database of this host to other hosts in your grid.

To delete a host, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Expand the **Grid Model** option, expand the **Hosts** options, right-click the **host that you want to delete**, then select **Delete**.

Figure 2–50   Select Delete
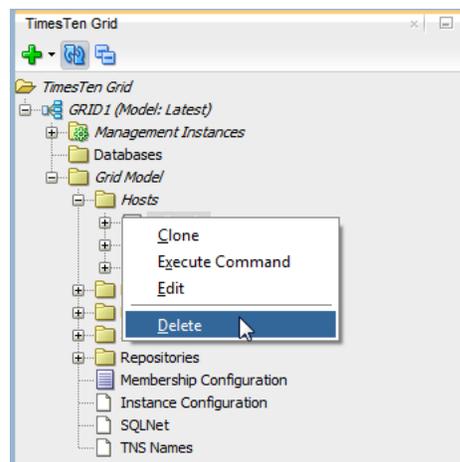
The Delete Host warning dialog displays. You see a warning message that asks you to ensure that you have migrated all the data from the database of this host to other hosts in your grid. Locate the Yes button.

2. Click **Yes**.

The Delete Host warning dialog closes and the Delete Host progress dialog displays. Deleting a host can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button in the Delete Host progress dialog.

3. Click **Close**.

You have successfully deleted a host. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

## Execute a command on a host

SQL Developer enables you to execute commands on hosts that are part of your grid. To execute a command on a host, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

> **Note:**   This procedure opens the Execute command dialog which enables you to select any host on which you want to execute commands. The Execute command dialog also enables you to execute a command on multiple hosts simultaneously.

1. Expand the **Grid Model** option, expand the **Hosts** options, right-click the **host on which you want to execute a command**, then select **Execute Command**.

Figure 2–51   Select Execute Command
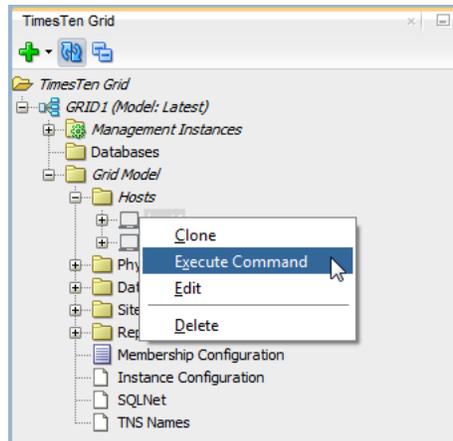
The Execute Command dialog displays. Locate the Excluded folder in the right part of the Execute Command dialog.

> **Note:**   If you want to see the data space group of each host, check the Show Data Space Group check box that is located at the bottom of the Execute Command dialog. This option separates the hosts in the Included and Excluded folders based on the data space group of each host.

2. In the **Excluded** folder, locate the host on which you want to execute a command. Right-click the **host name** and select **Move**. If you want to execute a command on additional hosts, right-click the corresponding host name and select Move for any additional hosts.

   Review the Included folder to ensure that the list contains the hosts on which you want to execute a command. Review the Excluded folder to ensure that the list contains the hosts on which you do not want to execute a command. Locate the Run Command text field.

3. In the **Run command** text field, type the command that you want to execute on the hosts that you selected.

   Locate the Execute button.

4. Click **Execute**.

Figure 2–52   Click Execute
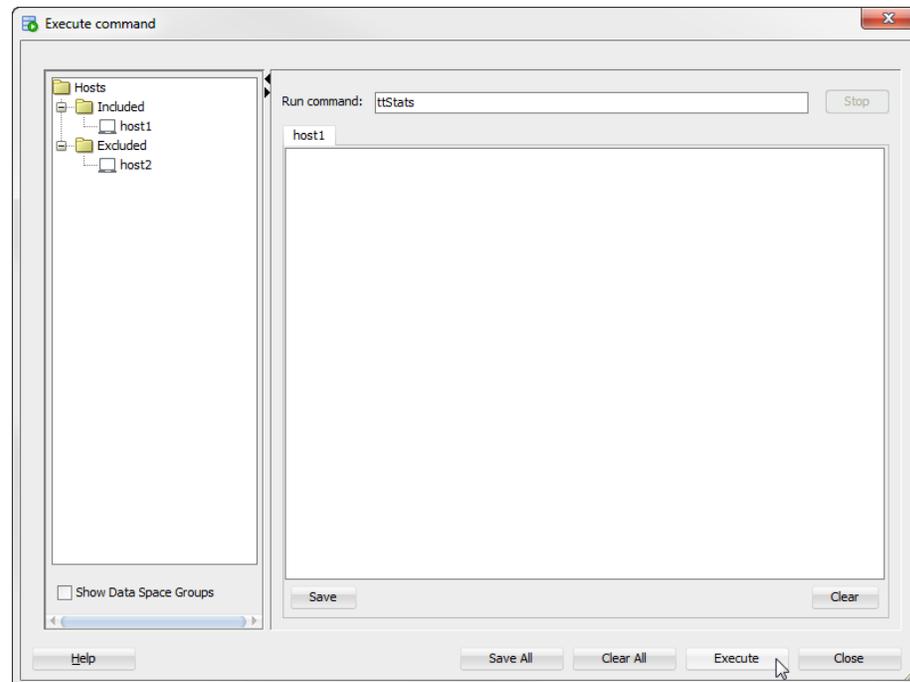
TimesTen Scaleout executes the command on all of the hosts that you selected. SQL Developer displays a tab with the output of every host. You can click on a host tab to view the output of the command for that specific host. You can save the output of a host by clicking the Save button. You can also save the output of every host by clicking the Save All button.

You have successfully executed a command on a host from your grid. You can execute additional commands by typing the new command in the Execute command text field and then clicking the Execute button. Once you have finished executing commands on your hosts, locate the Close button.

5. Click **Close**.

The Execute command dialog closes.

Working with installations

SQL Developer enables you to create or delete an installation from a host that is part of your grid. SQL Developer enables you to perform the following procedures with installations:

- Create a new installation

- Delete an installation

Create a new installation

SQL Developer enables you to create a new installation on a host that is part of your grid.

To create a new installation on a host, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Expand the **Grid Model** option, expand the **Hosts** options, expand the **host** on which you want to create an installation, right-click the **Installations** option, then select **New Installation**.

Figure 2–53   Select New Installation
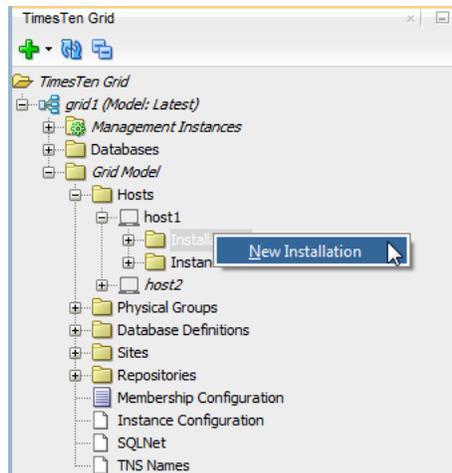
The New Installation dialog displays. Locate the Installation Name text field.

2. In the **Installation Name** field, type a name for your installation. Select a name that is a meaningful identifier. The default value is `installation`*n*, where *n* is the next sequential number based on your highest numbered installation name. For example, if you have an installation named `installation1`, the default value is `installation2`.

Locate the Select Source region of the New Installation dialog.

3. In the **Location** field, specify the directory on your selected host where the TimesTen installation file can be copied.

4. Select an installation source from the **Select Source** radio buttons. These installation sources are available:

   ■ Use existing source

   ■ New source

Use existing source

The Use Existing Source option enables you to copy an existing TimesTen installation from a host that is part of your grid. Before selecting the Use Existing Source option, ensure that you have completed the steps up to step 3 from "Create a new installation" on page 2-51.

1. From the **Host** drop-down list, select the host that contains the existing installation that you want to copy.

   The Installation drop-down list refreshes with the available installations of the selected host. If the host that you selected does not contain any TimesTen installation, SQL Developer switches your source to New Source.

2. From the **Installation** drop-down list, select the installation that you want to copy.

3. Click **OK**.

Figure 2–54   Click OK
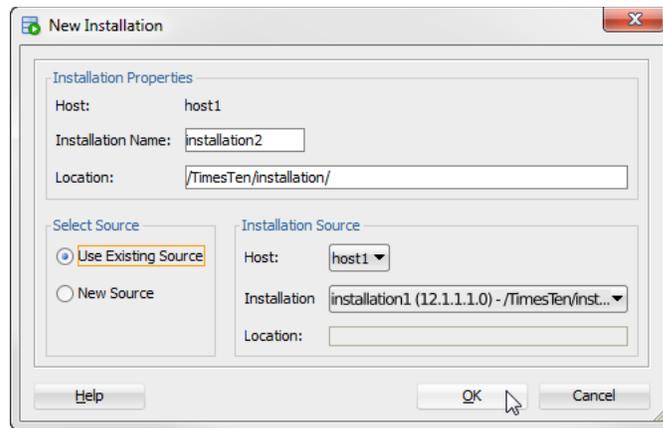
The New Installation dialog closes and the Creating Installation in grid model progress dialog displays. Creating a new installation can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button in the Creating Installation in grid model progress dialog.

4. Click **Close**.

You have successfully created a new TimesTen installation. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

New source

The New Source option enables you to use new installation files from a host that is part of your grid. Before selecting the New Source option, ensure that you have completed the steps up to step 3 from "Create a new installation" on page 2-51.

1. From the **Host** drop-down list, select the host that contains the new installation files that you want to copy.

2. In the **Location** field, specify the directory on your specified host where the TimesTen installation file are located.

3. Click **OK**.

Figure 2–55   Click OK
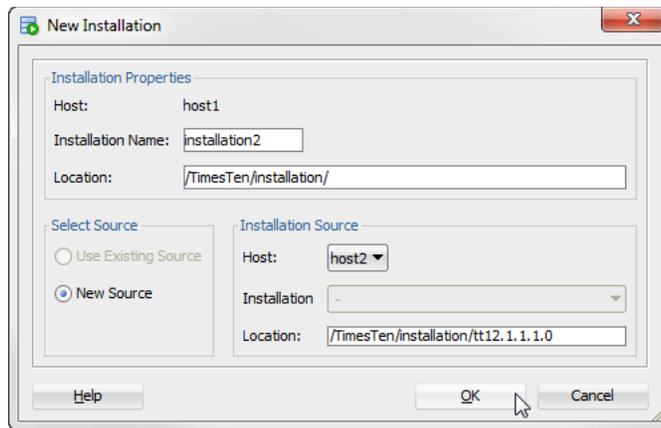
The New Installation dialog closes and the Creating Installation in grid model progress dialog displays. Creating a new installation can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button in the Creating Installation in grid model progress dialog.

4.   Click **Close**.

You have successfully created a new TimesTen installation. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

Delete an installation

SQL Developer enables you to delete an existing installation from a host that is part of your grid.

To delete a TimesTen installation from a host, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1.   Expand the **Grid Model** option, expand the **Hosts** options, expand the **host** from which you want to delete an installation, expand the **Installations** option, then right-click the installation and select **Delete**.

Figure 2–56   Select Delete
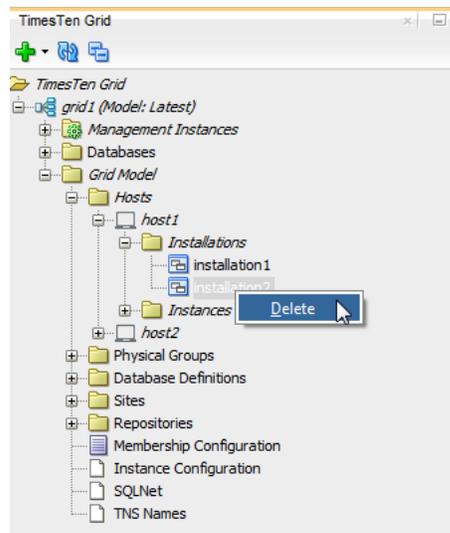
The Delete Installation dialog displays. Locate the Yes button.

2. Click **Yes**.

The Delete Installation dialog closes and the Delete Installation progress dialog displays. Deleting an installation can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button.

3. Click **Close**.

You have successfully deleted a TimesTen installation. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

Working with management instances

TimesTen uses one or two management instances to manage TimesTen Scaleout: an active management instance and a standby management instance. When managing TimesTen Scaleout, you perform all management operations through the active management instance to modify the grid model.

When you create a grid through the Simple Grid Wizard or the New Grid option in SQL Developer, TimesTen only creates an active management instance. It is recommended that you add a standby management instance to guard against failure and provide high availability for the management of your grid.

SQL Developer enables you to perform these management instance procedures:

- Create a new management instance

- Replace a management instance

- Switch the active management instance

- Control the standby management instance

- Use the shell on the host of the active management instance

Create a new management instance

To create a new management instance, ensure that you are on the main SQL Developer page and that you have enabled the TimesTen Grid view.

1. Right-click the name of your grid, select **Grid Management**, then select **New Management Instance**.

Figure 2–57 Create a new management instance
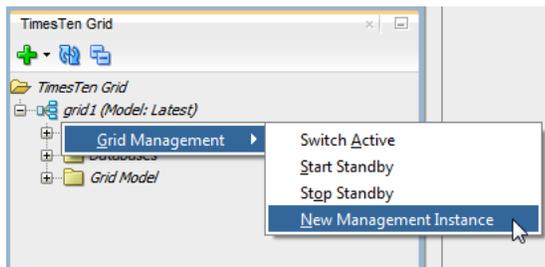
The New Management Instance dialog displays. Locate the Select Host region.

2. From the **Select Host** drop-down list, select the host where you want to create the management instance.

   If you want to add a new host, click the **New** button next to the Select Host drop-down list. See "Add a host" on page 2-42 for more information on the New Host dialog.

3. From the **Select Installation** drop-down list, select the installation that you want to use for the management instance. If you selected a host without an existing TimesTen installation, you need to create an installation on your selected host.

   If you want to create a new installation, click the **New** button next to the Select Installation drop-down list. See "Create a new installation" on page 2-51 for more information on the New Installation dialog.

4. In the **Instance Name** field, type a name for your management instance. Select a name that is a meaningful identifier. The default value is `instance1`.

5. In the **Instance Location** field, specify the directory where you want TimesTen to create the instance. TimesTen creates a directory with the instance name inside of the directory you specify. For example, if you have an instance name of `instance1` and an instance location of `/TimesTen/instance/`, TimesTen creates your instance in the `/TimesTen/instance/instance1/` directory.

6. In the **Daemon Port** field, specify the port number for the TimesTen daemon.

   If you selected a host without a TimesTen instance, the default value for the TimesTen daemon port is `6624`.

   If you selected a host with a TimesTen instance, the default value is the daemon port of your original instance plus 100. For example, if your original instance has a daemon port of `6624`, the default value is `6724`.

7. In the **Client/Server Port** field, specify the port number for the TimesTen client/server.

   If you selected a host without a TimesTen instance, the default value for the TimesTen client/server port is `6625`.

If you selected a host with a TimesTen instance, the default value is the client/server port of your original instance plus 100. For example, if your original instance has a client/server port of 6625, the default value is 6725.

8. In the **Management Port** field, specify the management port number.

The management port is the value for the TCP/IP port number of the first management instance replication agent.

If you selected a host without a TimesTen instance, the default value for the management port is 3754.

If you selected a host with a TimesTen instance, the default value is the management port of your original instance plus 100. For example, if your original instance has a management port of 3754, the default value is 3854.

9. Click **OK**.

Figure 2–58   Click OK
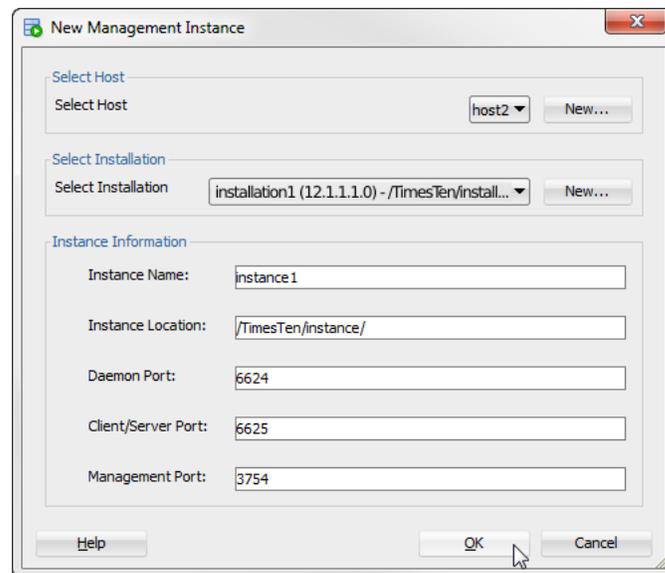
The Creating grid object(s) progress dialog displays. If your selected instance directory does not exist, a confirmation dialog displays to ensure that you want to create the new directory. If this warning dialog displays, click Yes.

Creating a management instance can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

The Management Instance dialog closes.

Locate the Close button.

10. Click **Close**.

You have successfully created a TimesTen management instance. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

## Replace a management instance

SQL Developer enables you to replace a management instance with a new instance. This is useful if you want to retire the host of an existing management instance and replace it with a new host.

You can only replace a management instance when your grid has an active and standby management instance.

If you replace the active management instance, TimesTen Scaleout promotes the standby management instance to the active management instance. Then, TimesTen Scaleout deletes your original active management instance and adds your new management instance as the standby management instance.

If you replace the standby management instance, TimesTen Scaleout deletes your current standby management instance and adds your new management instance as the standby management instance.

To replace an instance, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Expand the **Management Instances** option, right-click the management instance, then select **Replace**.

Figure 2–59   Select Replace
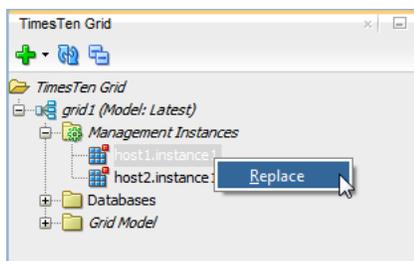
The Replace Management Instance dialog displays.

2. From the **Select Installation** drop-down list, select the installation that you want to use for the management instance. If you selected a host without an existing TimesTen installation, you need to create an installation on your selected host.

   If you want to create a new installation, click the **New** button next to the Select Installation drop-down list. See "Create a new installation" on page 2-51 for more information on the New Installation dialog.

3. In the **Instance Name** field, type a name for your management instance. Select a name that is a meaningful identifier. The default value is instance1.

4. In the **Instance Location** field, specify the directory where you want TimesTen to create the instance. TimesTen creates a directory with the instance name inside of the directory you specify. For example, if you have an instance name of instance1 and an instance location of /TimesTen/instance/, TimesTen creates your instance in the /TimesTen/instance/instance1/ directory.

5. In the **Daemon Port** field, specify the port number for the TimesTen daemon.

   If you selected a host without a TimesTen instance, the default value for the TimesTen daemon port is 6624.

   If you selected a host with a TimesTen instance, the default value is the daemon port of your original instance plus 100. For example, if your original instance has a daemon port of 6624, the default value is 6724.

6. In the **Client/Server Port** field, specify the port number for the TimesTen client/server.

If you selected a host without a TimesTen instance, the default value for the TimesTen client/server port is 6625.

If you selected a host with a TimesTen instance, the default value is the client/server port of your original instance plus 100. For example, if your original instance has a client/server port of 6625, the default value is 6725.

**7.** In the **Management Port** field, specify the management port number.

The management port is the value for the TCP/IP port number of the first management instance replication agent.

If you selected a host without a TimesTen instance, the default value for the management port is 3754.

If you selected a host with a TimesTen instance, the default value is the management port of your original instance plus 100. For example, if your original instance has a management port of 3754, the default value is 3854.

8. Click **OK**.

Figure 2–60   Click OK
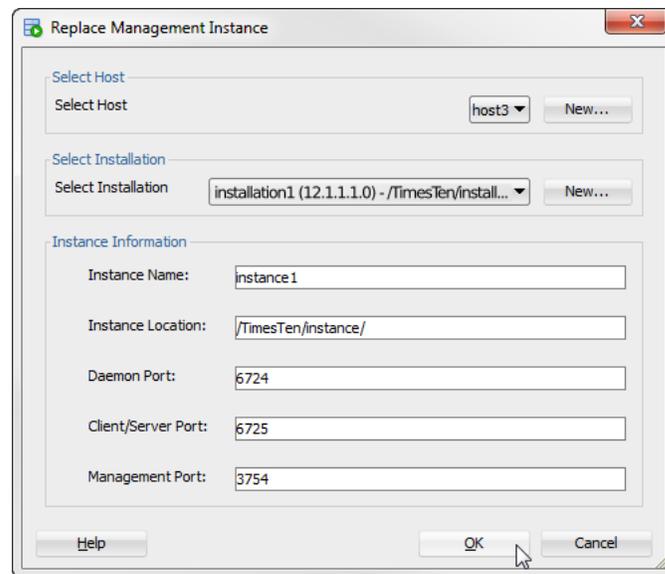
The Replace Management Instance progress dialog displays. If your selected instance directory does not exist, a confirmation dialog displays to ensure that you want to create the new directory. If this warning dialog displays, click Yes.

Replacing a management instance can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

The Replace Management Instance dialog closes.

Locate the Close button.

9. Click **Close**.

You have successfully created a new management instance, deleted an old management instance, and set the new management instance as your standby management instance. However, you need to apply your changes to the grid.

Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

Switch the active management instance

SQL Developer enables you to switch your current standby management instance to the active management instance of your grid.

Before you attempt to switch the active management instance, ensure that you have two management instances for your grid.

To switch the active management instance, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Right-click the name of your grid, select **Grid Management**, then select **Switch Active**.

Figure 2–61   Select Switch Active
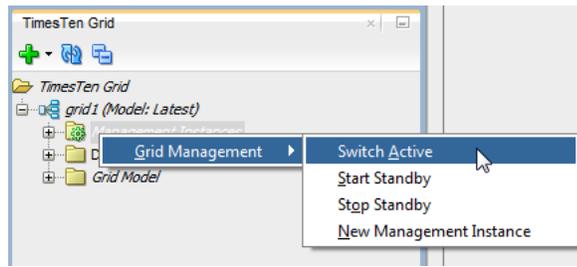
The Switch Active Management Instance progress dialog displays. Checking for a standby management instance can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

The Switch Active Management Instance progress dialog closes and the Switch Active Management Instance confirmation dialog displays. Locate the Yes button.

2. Click **Yes**.

The Switch Active Management Instance confirmation dialog closes and the Switch Active Management Instance progress dialog displays.

Switching the active management instance can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button.

3. Click **Close**.

You have successfully switched the active management instance. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

Ensure that you start your new standby management instance after you switch the active management instance. See "Control the standby management instance" on page 2-61 for more information.

Control the standby management instance

SQL Developer enables you to start or stop the standby management instance.

To start or stop the standby management instance, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Right-click **Management Instances**, select **Grid Management**, then select **one of these options**:

   ■ **Start Standby** - This option starts the standby management instance.

   ■ **Stop Standby** - This option stops the standby management instance.

Figure 2–62 Select an option
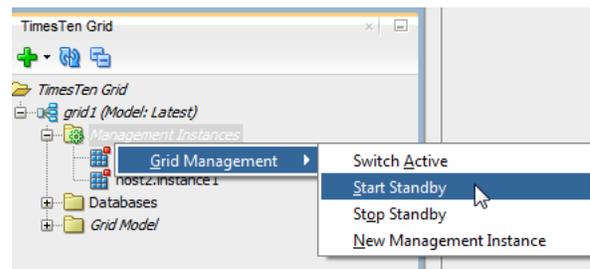
A confirmation dialog displays. Locate the Yes button.

If you attempt to start a standby management instance that is already started, a progress dialog displays with an error message. This also applies if you attempt to stop a standby management instance that is already stopped.

2. Click **Yes**.

The confirmation dialog closes and a progress dialog displays. Starting or stopping a standby management instance can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button.

3. Click **Close**.

You have successfully started or stopped a standby management instance.

Use the shell on the host of the active management instance

SQL Developer enables you to open a shell terminal to execute commands on the host of your active management instance. This is useful to execute certain `ttGridAdmin` utility commands that might not be supported by SQL Developer. For more information on `ttGridAdmin` commands, see "ttGridAdmin" in the *Oracle TimesTen In-Memory Database Reference*.

To start the shell terminal on the host of your active management instance, ensure that you are on the main SQL Developer page and that you have enabled the TimesTen Grid view.

Right-click the **name of your grid**, select **Execute**, then select **Shell**.

Figure 2–63   Select Shell
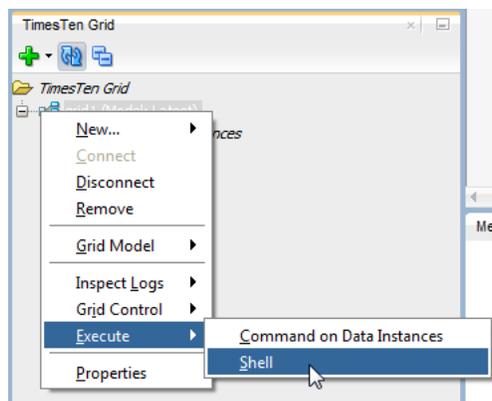
SQL Developer displays a tab with a shell terminal. You can execute commands on the host of your active management instance through this shell terminal.

Working with data instances

SQL Developer enables you to create or delete a data instance from a host that is part of your grid. SQL Developer enables you to perform the following procedures with instances:

- Create a new data instance
- Start or stop the instance and server
- Edit a data instance
- Execute a command on the host of a data instance
- Delete a data instance

Create a new data instance

SQL Developer enables you to create a new data instance with a data instance wizard or directly on a specific host.

Consider the following before creating a new data instance:

- Ensure that the hosts on which you create data instances for database distribution are assigned to a data space group. If you attempt to create a data instance on a host that is not assigned to a data space group, TimesTen Scaleout does not create that data instance. See "Assign hosts to data space groups" on page 2-11 for more information on assigning a host to a data space group.

Use one of these options to create a data instance:

- New data instance wizard - This option enables you to specify the host on which the instance is created.
- New data instance on a specific host - This option enables you to select the host on which the instance is created before opening the dialog.

New data instance wizard

To create a new data instance, ensure that you are on the main SQL Developer page and that you have enabled the TimesTen Grid view.

Right-click the name of your grid, then select **New Data Instance**.

Then follow the steps, starting on step 2, as described in "Create a new management instance" on page 2-56. You do need to specify a management port for a data instance.

New data instance on a specific host

SQL Developer enables you to create a new data instance on a host that is part of your grid.

To create a new data instance on a host, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Expand the **Grid Model** option, expand the **Hosts** options, expand the **host** on which you want to create an instance, right-click the **Instances** option, then select **New Instance**.

Figure 2–64   Select New Instance
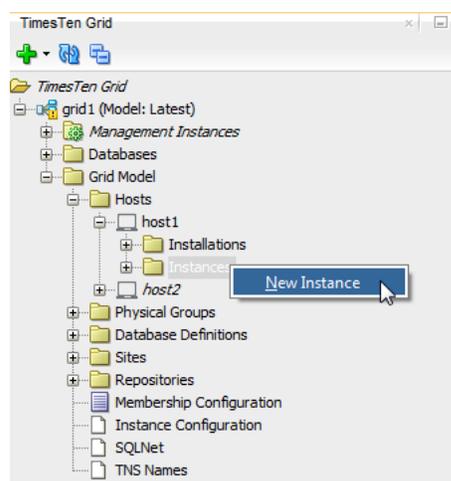
The New Instance dialog displays. Locate the Instance Name text field.

2. In the **Instance Name** field, type a name for your instance. Select a name that is a meaningful identifier. The default value is instance*n*, where *n* is the next sequential number based on your highest numbered instance name. For example, if you have an instance named instance1, the default value is instance2.

3. In the **Instance Location** field, specify the directory where you want TimesTen to create the instance. TimesTen creates a directory with the instance name inside of the directory you specify. For example, if you have an instance name of instance1 and an instance location of /TimesTen/instance/, TimesTen creates your instance in the /TimesTen/instance/instance1/ directory.

4. In the **Daemon Port** field, specify the port number for the TimesTen daemon. By default, the default value is the daemon port of your original instance plus 100. For example, if your original instance has a daemon port of 6624, the default value is 6724.

5. In the **Client/Server Port** field, specify the port number for the TimesTen client/server. By default, the default value is the client/server port of your original instance plus 100. For example, if your original instance has a client/server port of 6625, the default value is 6725.

6. From the **Installation** drop-down list, select the installation that you want TimesTen to use to create the instance. This drop-down list only shows available installations on the host on which you are creating the new instance.

To use an installation that is located on a different host or use a new installation source, click the New button that is located next to the Installation drop-down list. See "Create a new installation" on page 2-51 for more information about the New Installation dialog.

7. Click **OK**.

Figure 2–65   Click OK
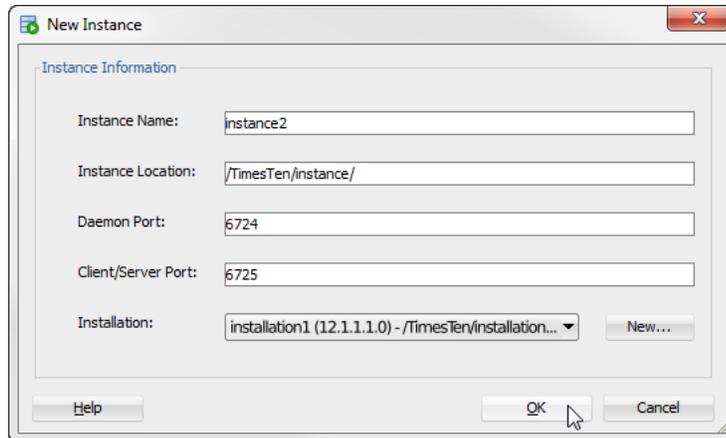
The New Instance dialog closes and the New Instance progress dialog displays. If your selected instance directory does not exist, a confirmation dialog displays to ensure that you want to create the new directory. If this warning dialog displays, click Yes.

Creating an instance can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button.

8. Click **Close**.

You have successfully created a TimesTen instance. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

Edit a data instance

SQL Developer enables you to edit the installation of a data instance.

> **Note:**   You can only edit a TimesTen data instance. You cannot edit a management instance.

To edit a data instance, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Expand the **Grid Model** option, expand the **Hosts** options, expand the **host** on which your instance is located, expand the **Instances** option, then right-click the instance and select **Edit**.

Figure 2–66   Select Edit
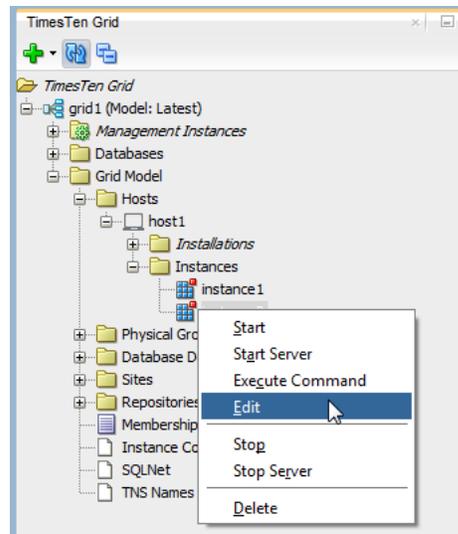
The Edit Instance dialog displays.

2. From the **Installation** drop-down list, select the new installation that you want to use for your instance.

3. Click **OK**.

   The Edit Instance dialog closes and the Edit Instance progress dialog displays. Editing an instance can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

   Locate the Close button in the Edit Instance progress dialog.

4. Click **Close**.

   The Edit Instance progress dialog closes.

You have successfully edited an instance. However, you need to apply your changes to the grid and restart your instance. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 and "Start or stop the instance and server" on page 2-65 for more information on applying model changes to a grid and restarting an instance, respectively.

Start or stop the instance and server

SQL Developer enables you to start or stop the TimesTen instance and the TimesTen server.

To start or stop the TimesTen instance or the TimesTen server, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Expand the **Grid Model** option, expand the **Hosts** options, expand the **host** on which your instance is located, expand the **Instances** option, then right-click the instance and select **one of these options**:

- **Start** - This option starts the TimesTen instance.

- **Start Server** - This option starts the TimesTen server.

- **Stop** - This option stops the TimesTen instance.

- **Stop Server** - This option stops the TimesTen server.

Figure 2–67 Select an option
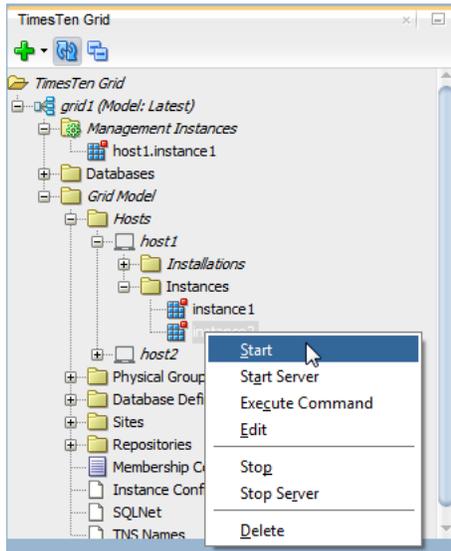
A confirmation dialog displays. Locate the OK button.

> **Note:** If you want to immediately start or stop a service and ignore warnings, select the **Force Action** check box.

2. Click **OK**.

The confirmation dialog closes. Starting or stopping a TimesTen instance or TimesTen server can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

Execute a command on the host of a data instance

SQL Developer enables you to execute commands on a particular data instance is located. To execute a command on a data instance, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

Right-click the **name of your grid**, select **Execute**, then select **Command on Data Instances**.

Figure 2–68   Select Execute Command on Data Instances
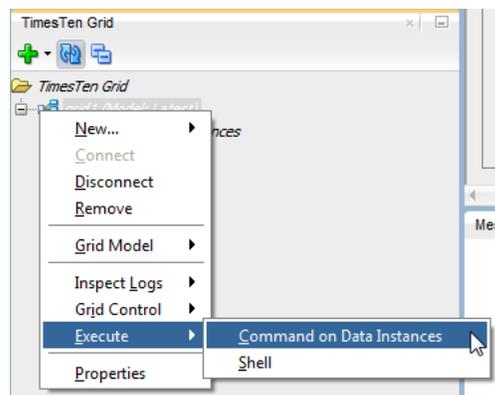
Alternatively, expand the **Grid Model** option, expand the **Hosts** options, expand the **host** on which your instance is located, expand the **Instances** option, then right-click the instance and select **Execute Command**.

Figure 2–69   Select Execute Command
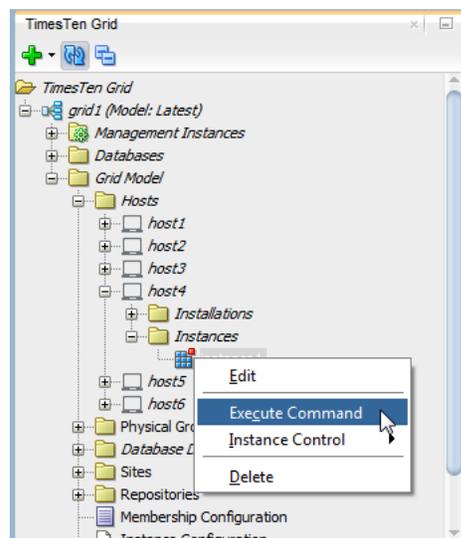
Then follow the steps, starting on step 2, as described in "Execute a command on a host" on page 2-49.

Delete a data instance

SQL Developer enables you to delete an existing data instance from a host that is part of your grid.

> **Note:**   You can only delete a stopped data instance. See "Start or stop the instance and server" on page 2-65 for more information on stopping a data instance.

Before you delete a TimesTen, ensure that you have migrated all data from databases of this instance to other TimesTen instances. For more information, see "Migrating,

Backing Up and Restoring Data" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

To delete a TimesTen instance from a host, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Expand the **Grid Model** option, expand the **Hosts** options, expand the **host** from which you want to delete an instance, expand the **Instances** option, then right-click the instance and select **Delete**.

Figure 2–70   Select Delete
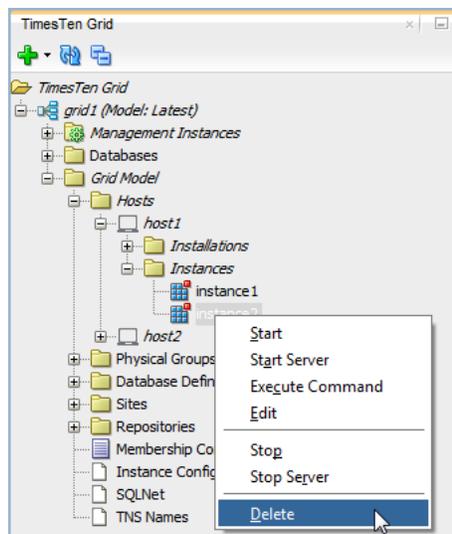
The Delete Instance dialog displays. Ensure that you have migrated all data from databases of this instance to other TimesTen instances.

Locate the Yes button.

2. Click **Yes**.

The Delete Instance dialog closes and the Delete Instance progress dialog displays. Deleting an instance can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button.

3. Click **Close**.

The Delete Instance progress dialog closes.

You have successfully deleted a TimesTen instance. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

Working with configuration files

SQL Developer enables you to import and export multiple configuration files for your grid. You can import and export these configuration files:

> **Note:** If you have not imported a file for a specific configuration, you cannot export a file for that configuration.

- Membership Configuration - This file contains configuration parameters for the membership service configuration for your grid. For more information on membership services, see "Overview of the TimesTen Scaleout membership service" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

- Instance Configuration - This file contains configuration parameters that are used by each instance in the grid. For more information on this file, see "Import instance configuration attributes (instanceConfigImport)" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

- SQLNet - This file contains configuration parameters for the grid to interact with an Oracle database through the `ttLoadFromOracle` utility. For more information on this file, see "Import a sqlnet file (SQLNetImport)" in the *Oracle TimesTen In-Memory Database Reference*.

- TNS Names - This file contains configuration parameters for the grid to interact with an Oracle database through the `ttLoadFromOracle` utility. For more information on this file, see "Import TNS names (TNSNamesImport)" in the *Oracle TimesTen In-Memory Database Reference*.

SQL Developer enables you to perform these operations with configuration files:

- Import a configuration file
- Export a configuration file

Import a configuration file

SQL Developer enables you to import multiple types of configuration files.

To import a configuration file, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Expand the **Grid Model** option, then right-click the **configuration file name** and select **Import**.

Figure 2–71   Select Import
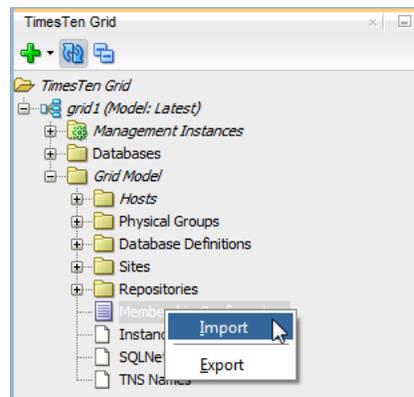
The Import dialog displays. Locate the Browse button.

2. Click **Browse**.

The Import file dialog displays. Browse for your configuration file.

3. Select **your configuration file** and click **Open**.

The Import file dialog closes. In the Import dialog, locate the Import button.

4. Click **Import**.

Figure 2–72   Click Import
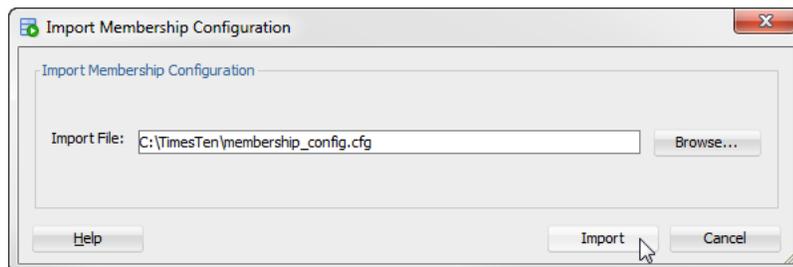
The Verifying File Location progress dialog displays. SQL Developer verifies that the configuration file that you specified is a valid file. If SQL Developer displays an error message, troubleshoot the error.

The Verifying File Location progress dialog and the Import Configuration dialog close. The Importing Configuration progress dialog displays. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button.

5. Click **Close**.

The Import Configuration progress dialog closes.

You have successfully imported a configuration file in the grid model. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

Export a configuration file

SQL Developer enables you to export multiple types of configuration files.

To export a configuration file, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Expand the **Grid Model** option, then right-click the **configuration file name** and select **Export**.

Figure 2–73   Select Export
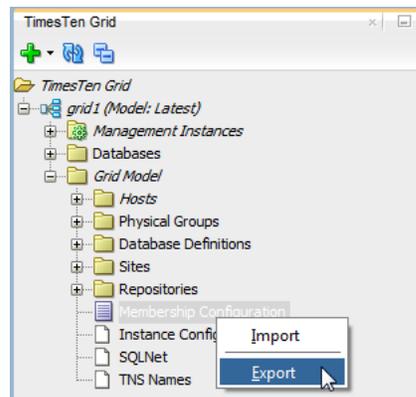
The Export dialog displays. Locate the Choose Version to Export region.

2. From the Choose Version to Export region, select **a version of the configuration to export**. These are the options:

    ■   **Current** - This is the current configuration file that is applied to the grid.

    ■   **Latest** - This is the latest configuration file that is not yet applied to the grid.

    ■   **Version** - This is the version of the configuration file that you can select. If you select this option, use the text field to the right of the Version radio button to select the version to export.

    Locate the Browse button.

3. Click **Browse**.

    The Export file dialog displays. Specify the location of where you want to export the configuration file.

4. In the **File Name** text field, specify a meaningful name for the **configuration file** and click **Save**.

    The Export file dialog closes. In the Export dialog, locate the Export button.

5. Click **Export**.

Figure 2–74   Click Export
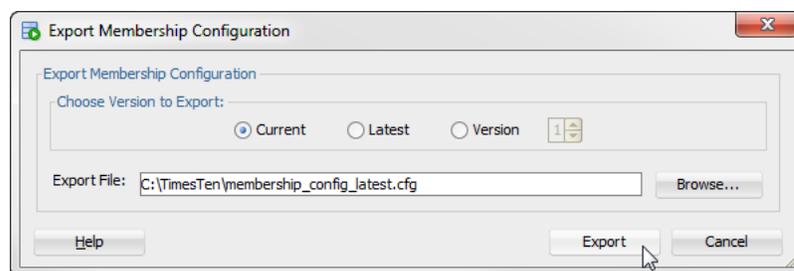
The Verifying File Location progress dialog displays. If SQL Developer displays an error message, troubleshoot the error.

The Verifying File Location progress dialog and the Export Configuration dialog close. The Export Configuration progress dialog displays. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button.

6. Click **Close**.

The Export Configuration progress dialog closes.

You have successfully exported a configuration file from the grid model.

Working with databases

SQL Developer enables you to create or delete a database from your grid.

Consider these things when you decide whether to use the database wizard or not:

- The database wizard automatically applies the grid model changes at the end of the wizard. If you want to verify the model changes before applying the grid model, create a database definition and then create your database from that definition.

SQL Developer enables you to perform the following procedures with databases:

- Create a database with the database wizard
- Working with database definitions
- Define the distribution map of a database
- Perform database operations
- Working with connectables
- View active connections to the database
- Destroy a database

Create a database with the database wizard

SQL Developer enables you to create a database in your grid with the database wizard. You can also create a database from a database definition. See "Working with database definitions" on page 2-78 for more information.

Before you create a database:

- Ensure that you have a functional grid. See "Create a grid" on page 2-5 and "Manage an existing grid" on page 2-17 for more information.
- Ensure that your grid has at least an equal amount of data instances to the K-safety level of your grid. For example if you have a K-safety level of 2, you need to have at least two data instances. See "Working with data instances" on page 2-62 for more information on adding data instances.
- Ensure that each of your data space groups are associated with an equal amount of data instances. Hosts, not data instances, are assigned to data space groups. It is recommended that you have a single data instance on each host. See "Assign hosts to data space groups" on page 2-11 for more information on assigning a host to a data space group.

To create a database, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Right-click the **Databases** option and then select **New Database**.

Figure 2–75   Select New Database
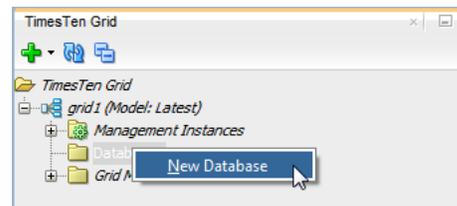
The TimesTen Grid Database Wizard dialog and Comparing current and latest grid model progress dialog displays.

If you have made changes to your grid model and have not applied the model changes, the Comparing current and latest grid model progress dialog closes and a Changes in current grid model dialog displays. This dialog displays objects that have been added, changed, or removed from the grid model. Creating a database involves applying the grid model changes, so ensure that you want these grid model changes to be applied. Locate the OK button of the Changes in current grid model dialog.

See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

If you have not made changed to your grid model, locate the Close button of the Comparing current and latest grid model progress dialog.

2. Depending on the dialog that displays, either click **OK** or **Close**.

Depending on the dialog that displayed, the Changes in current grid model dialog or the Comparing current and latest grid model progress dialog closes. The TimesTen Grid Database Wizard displays.

Locate the Database Name field.

3. In the **Database Name** text field, type a name for your database. Select a name that is a meaningful identifier. The default value is database1.

Note: Optionally, you can import a database definition file to load attribute values for your database definition. The database definition file must use .dbdef as the file name extension.

To import a database definition file, click the **Import Definition** button. Then, select **your database definition file** and click **Open**. If an attribute value was set in the wizard before you imported the database definition file, the attribute value changes to the value that is defined in the database definition file.

Once you have imported the database definition file, you can still edit the database definition within the wizard. If you do not want to keep editing the database definition, skip to step 5.

For more information on the database definition file, see "Creating a database definition file" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

If your grid already contains a database, the default value of the database name of your original database plus 1. For example, if your original database name is database1, the default value is database2.

4. Set the connection attributes for your database. You can set these connection attributes:

- **Data Store Attribute** - Attributes that are set at database creation time.

  > **Note:** Ensure that you set a valid path for the `DataStore` attribute. This path must be valid on every host for every data instance.
  >
  > It is best practice to specify `LogDir` and have it be a different location from `DataStore`. For additional considerations for database and user file locations, see "Planning installation and deployment" in the *Oracle TimesTen In-Memory Database Installation, Migration, and Upgrade Guide*.

- **First Connection Attribute** - Attributes that are set when a database is loaded into memory.

- **General Connection Attribute** - Attributes that are set by each connection and persist for the duration of the connection. These attributes are added to a connectable and not are part of a database definition.

To set any attribute that is part of the type of connection attributes listed above, select the relevant tab. Each tab of connection attributes contains a list of attributes with default values. You can edit the value of an attribute by double clicking the value from the value column of an attribute. Depending on the connection attribute, you can edit the value with a text field or a drop-down list of valid values.

You can add additional connection attributes by selecting a connection attribute from the drop-down list located below the connection attribute tabs and then clicking **Add**. If a connection attribute is not listed in the drop-down list, you can select the **New Attribute** option and enter the connection attribute name in the Attribute column of the connection attribute table.

You can remove a connection attribute by selecting a connection attribute from the connection attribute table and then clicking **Delete**.

For more information on valid connection attributes and values, see "List of Attributes" in the *Oracle TimesTen In-Memory Database Reference*.

5. Once you have set your connection attributes, click **Next**.

Figure 2–76   Click Next
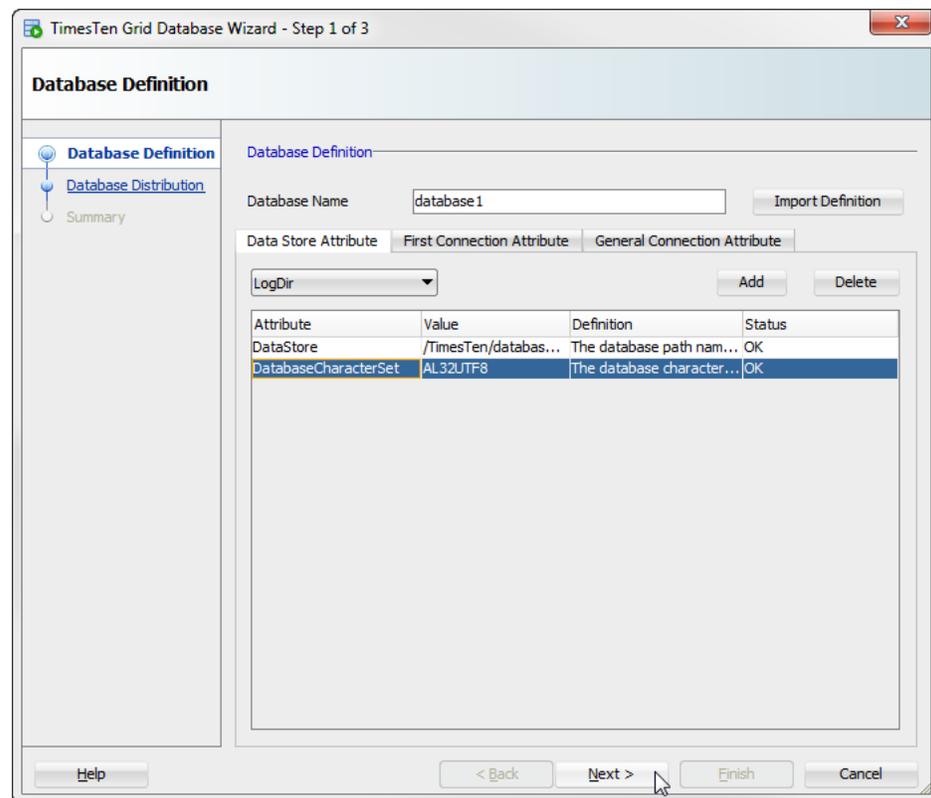
SQL Developer validates your connection attribute values. If SQL Developer displays an error message, troubleshoot the error and click the Next button again to revalidate your input values.

The Checking Directory Attributes dialog displays. Checking for a directory can take a few minutes. If your selected database directory does not exist on the host of a data instance, a confirmation dialog displays to ensure that you want to create the new directory on that host. If this warning dialog displays, click Yes.

> **Note:** SQL Developer creates the selected database directory on all hosts of available data instances.

The Checking Directory Attributes dialog closes and the TimesTen Grid Database Wizard dialog refreshes. Locate the Database Distribution map.

6. By default, the default value is the maximum number of data instances based on their data space group without leaving a replica set incomplete. Review the SQL Developer suggestion for data instances and perform any necessary modifications.

> **Note:** If you do not want to configure the distribution map in this wizard, check the **Skip database distribution** check box and click **Next**. See "Define the distribution map of a database" on page 2-85 for more information on defining a distribution map for a database.

> **Note:** Ensure that each data space group contains the same number of data instances to create complete replica sets. If your data space groups do not contain the same number of data instances, an error dialog displays after you click the Apply button.

If you want to remove a suggested data instance, right-click the **instance name** and select **Remove**.

If you want to add a data instance from the Available column, right-click the **instance name** and select **Add**.

7. Once you have selected all of your data instances for data distribution, click **Next**.

Figure 2–77   Click Next
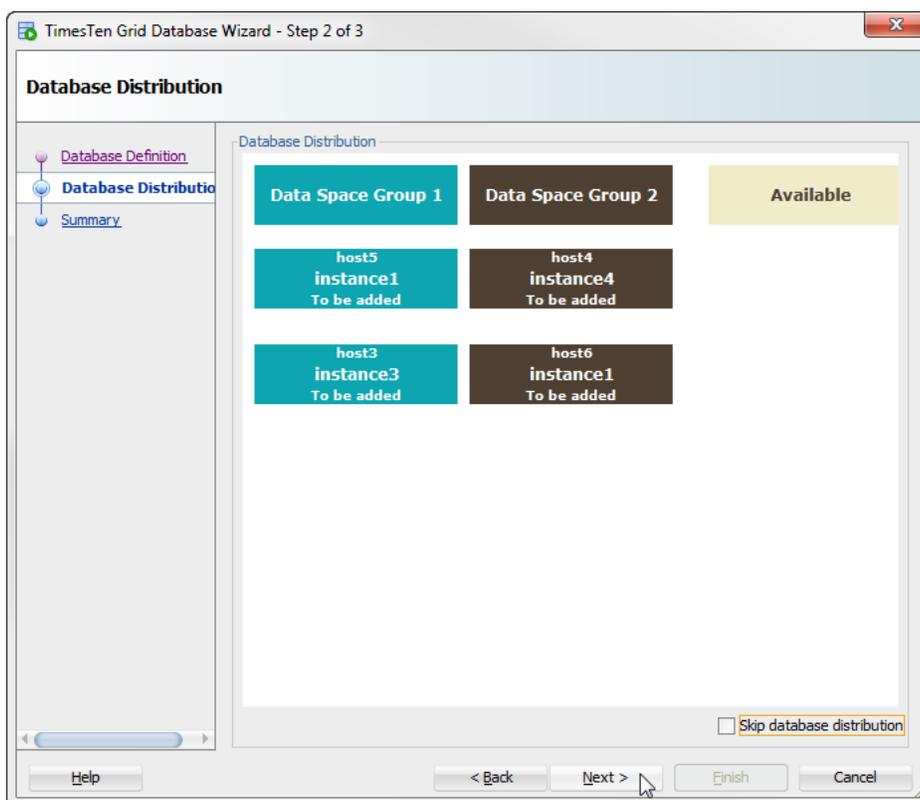
The TimesTen Grid Database Wizard refreshes and shows a summary table with the configuration for your database.

8. Verify that the configuration settings for your database are correct and click **Finish**.

> **Note:** Optionally, click the **Export Definition** button to save the database definition as a `.dbdef` file.

Figure 2–78   Click Finish
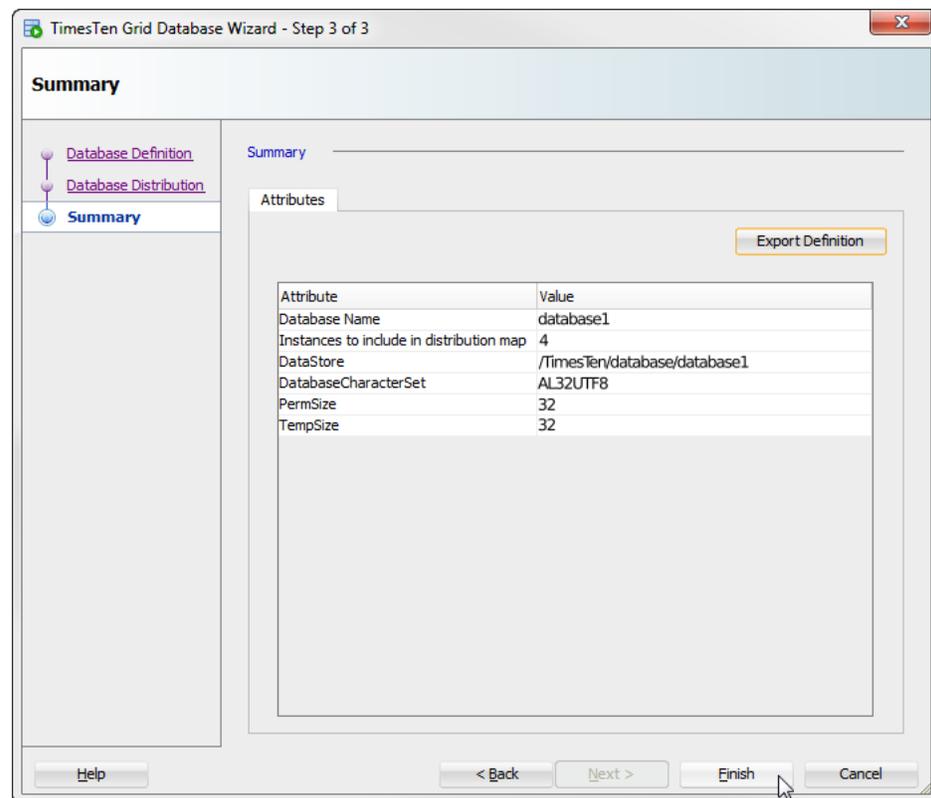
The TimesTen Grid Database Wizard dialog closes and a Creating Database progress dialog displays. Creating a database can take a few minutes depending on the number of data instances included in the distribution map. If SQL Developer displays an error message, troubleshoot the error. In the case that an error file is generated, the Creating Database progress dialog displays the error file as a hyperlink. You can save the error file and troubleshoot any errors that are listed in the file.

> **Note:** If SQL Developer displays an error or if you cancel the database creation process, review the grid navigator to see if the database and definition file were created. In case that a database or database definition was created, delete the database and database definition. Then, create a new database. See "Destroy a database" on page 2-95 for more information on deleting a database.

9. Click **Close**.

The Creating Database progress dialog closes.

You have successfully created a database in your grid.

Before you attempt to connect an application to the database, ensure that you open the database and create a user with the appropriate privileges to connect to the database. To create a user for your database with SQL Developer, use the `ttIsql` utility through the Execute command on a host dialog. You can execute this utility on any of the data instances of your database.

This command creates a user, `terry`, with a password of `terry` for the database, `database1`. Then, `terry` is granted `admin` privileges.

```
ttisql -e 'CREATE USER terry IDENTIFIED BY terry;
GRANT ADMIN TO terry; ' DSN=database1
```

See "Perform database operations" on page 2-88 and "Execute a command on a host" on page 2-49, for more information on opening a database and executing a command on a host, respectively.

See "Managing Access Control" in the *Oracle TimesTen In-Memory Database Security Guide* for more information on creating users.

Working with database definitions

A database definition contains the description of a TimesTen Scaleout database. It defines the database name, as well as the attributes associated with the database. Once a database definition is added to the current version of the model, it can be used to create a database.

SQL Developer enables you to perform the following procedures with database definitions:

- Create a database definition
- Create a database from a database definition
- Edit a database definition
- Export a database definition
- Delete a database definition

Create a database definition

SQL Developer enables you to create a database definition. To create a database definition, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

> **Note:** When you create a new database definition, TimesTen also creates a direct connectable for this new database definition.

1. Expand the **Grid Model** option, right-click the **Database Definitions** option, then select **New Database Definition**.

Figure 2–79   Select New Database Definition
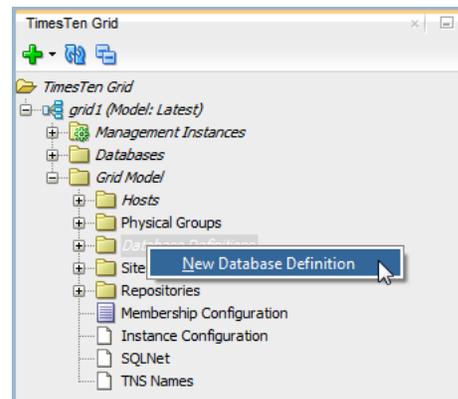
The New Database Definition dialog displays. Locate the Database Name field.

2. In the **Database Name** text field, type a name for your database. Select a name that is a meaningful identifier. The default value is `database1`.

> **Note:**   Optionally, you can import a database definition file to load attribute values for your database definition. The database definition file must use `.dbdef` as the file name extension.
>
> To import a database definition file, click the **Import Definition** button. Then, select **your database definition file** and click **Open**. If an attribute value was set in the wizard before you imported the database definition file, the attribute value changes to the value that is defined in the database definition file.
>
> Once you have imported the database definition file, you can still edit the database definition within the wizard. If you do not want to keep editing the database definition, skip to step 4.
>
> For more information on the database definition file, see "Creating a database definition file" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

If your grid already contains a database, the default value is the database name of your original database plus 1. For example, if your original database name is `database1`, the default value is `database2`.

3. Set the connection attributes for your database definition. You can set these connection attributes:

    ■   **Data Store Attribute** - Attributes that are set at database creation time.

    > **Note:**   Ensure that you set a valid path for the `DataStore` attribute. This path must be valid on every host for every data instance.
    >
    > It is best practice to specify `LogDir` and have it be a different location from `DataStore`. For additional considerations for database and user file locations, see "Planning installation and deployment" in the Oracle TimesTen In-Memory Database Installation, Migration, and Upgrade Guide.

- **First Connection Attribute** - Attributes that are set when a database is loaded into memory.

- **General Connection Attribute** - Attributes that are set by each connection and persist for the duration of the connection.

To set any attribute that is part of the type of connection attributes listed above, select the relevant tab. Each tab of connection attributes contains a list of attributes with default values. You can edit the value of an attribute by double clicking the value from the value column of an attribute. Depending on the connection attribute, you can edit the value with a text field or a drop-down list of valid values.

You can add additional connection attributes by selecting a connection attribute from the drop-down list located below the connection attribute tabs and then clicking **Add**. If a connection attribute is not listed in the drop-down list, you can select the **New Attribute** option and enter the connection attribute name in the Attribute column of the connection attribute table.

You can remove a connection attribute by selecting a connection attribute from the connection attribute table and then clicking **Delete**.

For more information on valid connection attributes and values, see "List of Attributes" in the *Oracle TimesTen In-Memory Database Reference*.

4. Once you have set your connection attributes, click **OK**.

Figure 2–80   Click OK
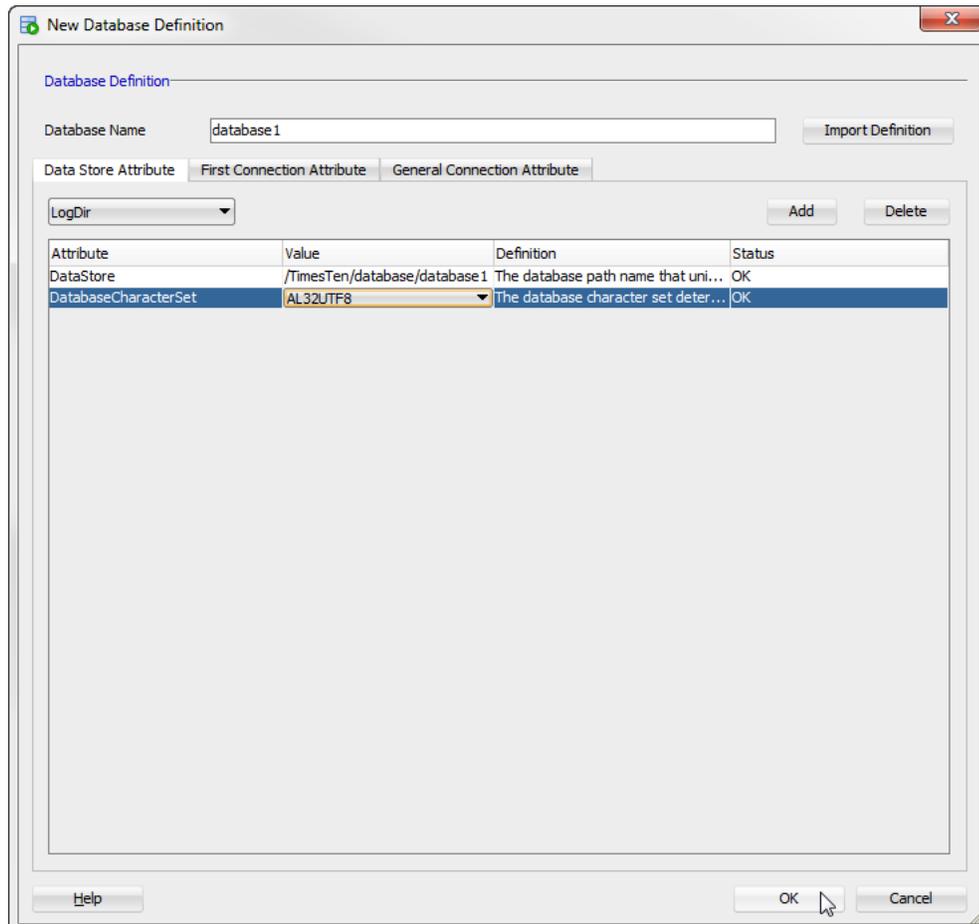
SQL Developer validates your connection attribute values. If SQL Developer displays an error message, troubleshoot the error and click the OK button again to revalidate your input values.

The Checking Directory Attributes dialog displays. Checking for a directory can take a few minutes. If your selected database directory does not exist on the host of a data instance, a confirmation dialog displays to ensure that you want to create the new directory on that host. If this warning dialog displays, click Yes.

> **Note:** SQL Developer creates the selected database directory on all hosts of available data instances.

The Checking Directory Attributes dialog closes and the Creating Database Definition progress dialog displays. Creating a database definition can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button.

5. Click **Close**.

The Creating Database Definition progress dialog closes.

You have successfully created a database definition and a direct connectable. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

> **Note:** After creating a database definition, a database icon from your database definition displays under the Databases node of your grid node in SQL Developer. However, you are unable to perform database operation with this database until you create the database from your database definition. See "Create a database from a database definition" on page 2-81 for more information.

Create a database from a database definition

SQL Developers enables you to create a database from a database definition. To create a database from a database definition, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

If the database definition is already used for another database, SQL Developer displays an error message. You can only create a database from a database definition that has not been used to create another database.

1. Expand the **Grid Model** option, expand the **Database Definitions** option, right-click the **name of your database definition**, then select **Create Database**.

Figure 2–81 Select Create Database
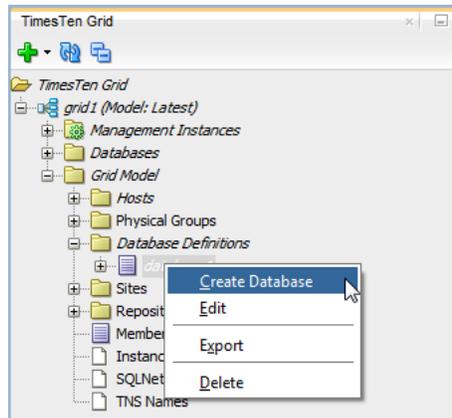
The Command issued dialog displays and a Creating Database progress dialog displays in the lower right corner of the main SQL Developer page. Locate the OK button of the Command issue dialog.

2. Click **OK**.

The Command issued dialog closes. Creating a database can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

> **Note:** If SQL Developer displays an error or if you cancel the database creation process, review the grid navigator to see if the database and definition file were created. In case that a database or database definition was created, delete the database and database definition. Then, create a new database. See "Destroy a database" on page 2-95 for more information on deleting a database.

You have successfully created a database from a database definition. You are now ready to specify the distribution map of the database.

Before you attempt to connect an application to the database, ensure that you open the database and create a user with the appropriate privileges to connect to the database. To create a user for your database with SQL Developer, use the ttIsql utility through the Execute command on a host dialog. You can execute this utility on any of the data instances of your database.

This command creates a user, terry, with a password of terry for the database, database1. Then, terry is granted CREATE SESSION privileges.

```
ttisql -e 'CREATE USER terry IDENTIFIED BY terry;
GRANT CREATE SESSION TO terry; ' DSN=database1
```

See "Perform database operations" on page 2-88 and "Execute a command on a host" on page 2-49, for more information on opening a database and executing a command on a host, respectively.

See "Managing Access Control" in the *Oracle TimesTen In-Memory Database Security Guide* for more information on creating users.

Edit a database definition

SQL Developer enables you to edit an existing database definition.

Before you edit an existing database definition:

- If you want to edit a database attribute and there is a database that was created from the database definition you are editing, you must stop the database, destroy it, and recreate it with the new database attributes. Since you are destroying the database, this causes you to lose all data in that database.

- If you want to edit a general connection attribute, edit the respective connectable because general connection attributes are associated with a connectable. See "Edit a connectable" on page 2-92 for more information.

- If you edit a first connection attribute of a database definition of an open database, these changes are not applied until you unload and reload the database. SQL Developer displays a warning message if your database is currently open when you edit the database definition. See "Perform database operations" on page 2-88 for more information on closing and opening a database.

To edit a database definition, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

Expand the **Grid Model** option, expand the **Database Definitions** option, right-click the **name of your database definition**, then select **Edit**.

Figure 2–82   Select Edit
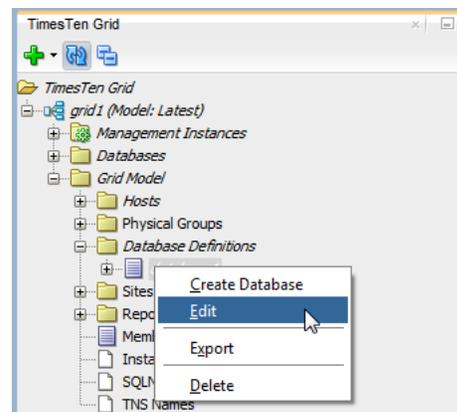
The Edit Database Definition dialog and the Checking if database exists progress dialog display.

If there is an open database that was created from the database definition you are editing, the Checking if database exists progress dialog closes and a Editing open database warning dialog displays. If you do not want to see the Editing open database warning for future operations, select the **Skip This Message Next Time** check box. Then, click the **OK** button.

If there is no open database that was created from the database definition you are editing, the Checking if database exists progress dialog closes.

Follow the steps, starting on step 3, as described in "Create a database definition" on page 2-78.

Export a database definition

SQL Developer enables you to export an existing database definition. You can use this exported database definition file to import the definition of a new database definition.

To export a database definition, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

Expand the **Grid Model** option, expand the **Database Definitions** option, right-click the **name of your database definition**, then select **Export**.

Figure 2–83   Select Export
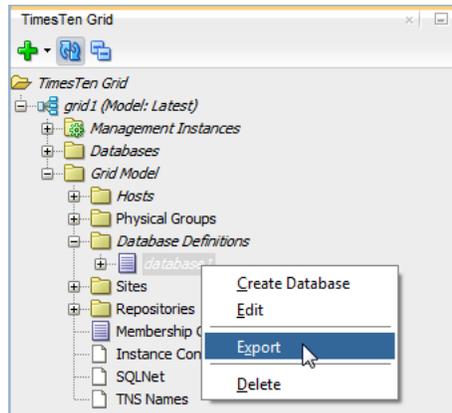
The Export Definition dialog displays.

Specify the location of where you want to export the connectable file.

In the **File Name** text field, specify a meaningful name for the **database definition file** and click **Save**.

The Export Definition dialog closes.

You have successfully exported a database definition file.

Delete a database definition

SQL Developer enables you to delete a database definition from your grid model.

You cannot delete a database definition if it references an existing database. You must destroy the database first. See "Destroy a database" on page 2-95 for more information.

To delete a database definition, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

Expand the **Grid Model** option, expand the **Database Definitions** option, right-click the **name of your database definition**, then select **Delete**.

Figure 2–84   Select Delete
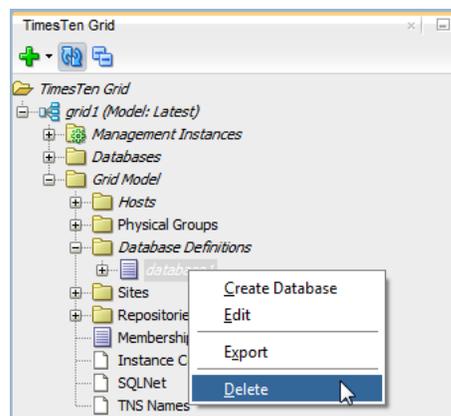
The Delete database definition dialog displays. Locate the Yes button.

Click **Yes**.

The Delete database definition dialog closes and the Delete database definition progress dialog displays. Locate the Close button.

Click **Close**.

You have successfully deleted a database definition. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

Define the distribution map of a database

SQL Developers enables you to define the distribution map of a new database from a database definition. You can also redefine the distribution map of an existing database. You cannot access a database that is in the process of defining or redefining its distribution map. To define the distribution map of a database, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

> **Note:** Redefining the distribution map of an existing database can take up to a few hours. Ensure that you do not need to access this database before redefining the distribution map.

1. Expand the **Databases** option, right-click the **name of your database**, then select **Redistribute**.

Figure 2–85   Select Redistribute
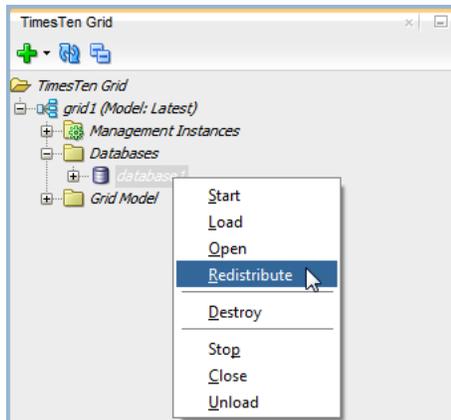
The Distribution for database dialog displays. Locate the Database Distribution map.

2. If you want to add a data instance from the Available column, right-click the **instance name** and select **Add**.

   If you want to remove a suggested data instance, right-click the **instance name** and select **Remove**. If you want to replace the data instance that you are removing, select **Remove and replace with** and select the new data instance.

   If you want to evict a data instance, right-click the **instance name** and select **Evict**. Only evict a data instance if you are unable to fix that data instance and you have a backup of the data from that data instance. Evicting a data instance can result in data loss. If you want to replace the data instance that you are evicting, Select **Evict and replace with** and select the new data instance.

   > **Note:**   Ensure that each data space group contains the same number of data instances to create complete replica sets. If your data space groups do not contain the same number of data instances, an error dialog displays after you click the Apply button.

3. Once you have selected all of your data instances for data distribution, click **Apply**.

Figure 2–86   Click Apply
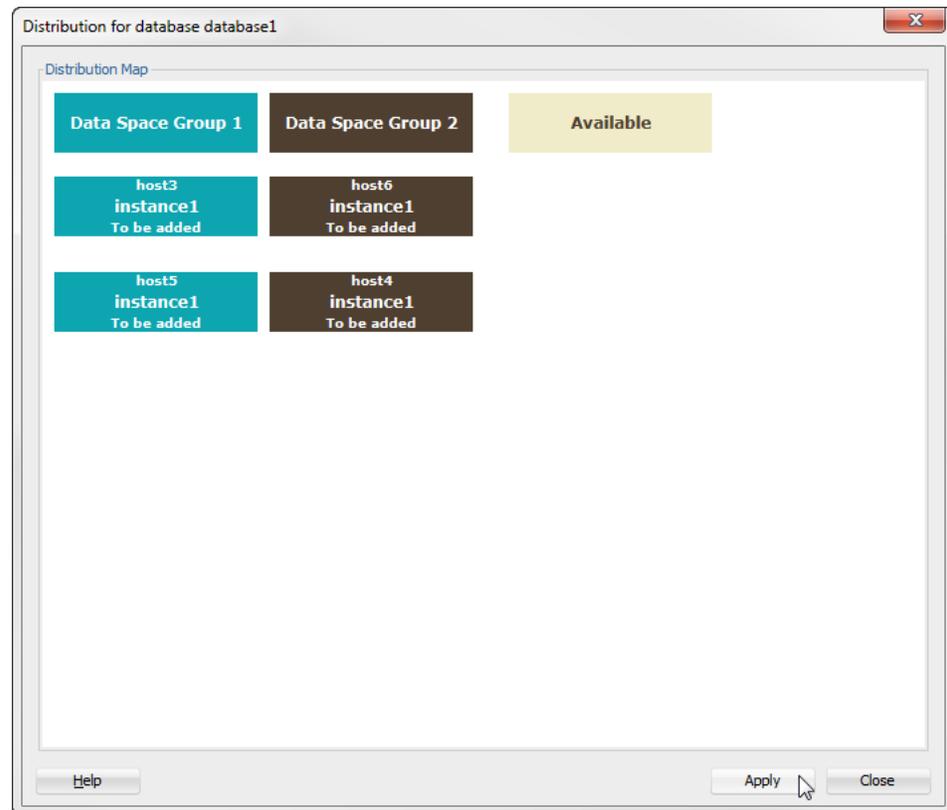
The Performing operations progress dialog displays. Defining the distribution map of a new database can take a few minutes. If you are redistributing the data of an existing database, the redistribution can take up to a few hours. If SQL Developer displays an error message, troubleshoot the error.

Once the distribution map of your database is applied, the Performing operations progress dialog closes and the Distribution for database dialog refreshes. The distribution map shows how your data instances are distributed in replica sets and data space group.

Locate the Close button.

4. Click **Close**.

Figure 2–87   Click Close
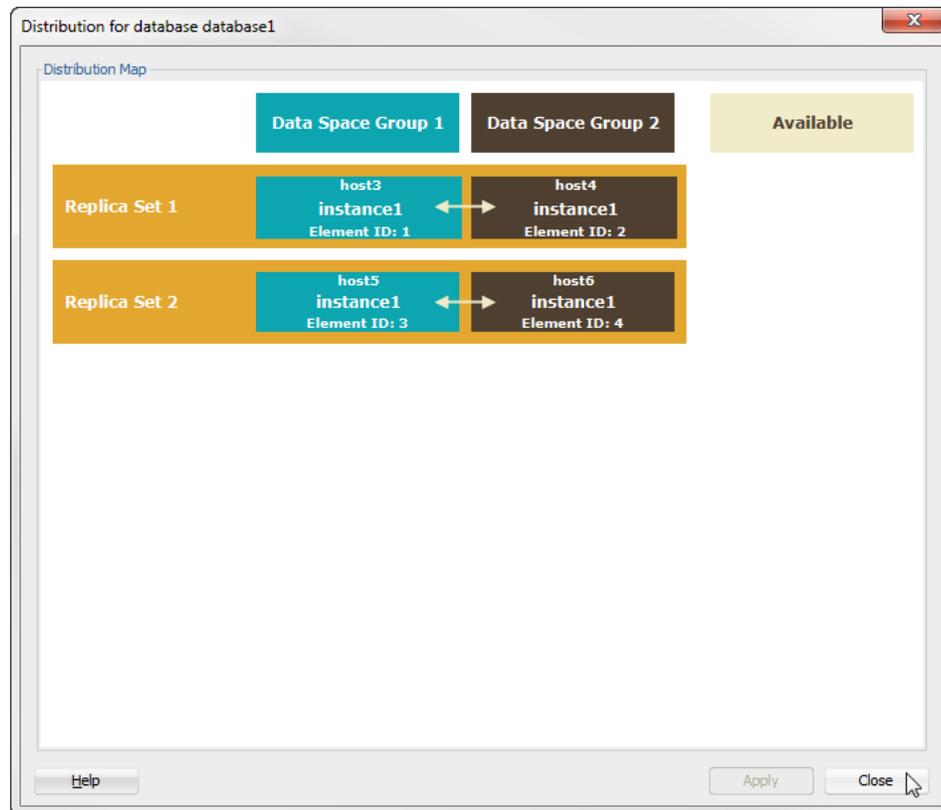
The Distribution for database dialog closes.

You have successfully defined the distribution map of a database.

Perform database operations

SQL Developer enables you to start, load, open, stop, close, and unload the TimesTen database.

To start, load, open, stop, close, or unload the TimesTen database, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

Expand the **Databases** option, then right-click the **name of your database** and select **one of these options**:

> **Note:**   Depending on the current state of your database, you may be unable to perform a certain operation. If SQL Developer is unable to perform a specific operation, an error or warning message displays. For example, you cannot open a database that is already open.

- **Start** - Loads every element of the database into memory of their respective hosts and enables the database for user connections. This option does the same thing as if you perform a load and open operation of the database.

- **Load** - Loads every element of the database into memory of their respective hosts.

- **Open** - Enables the database for user connections.

- **Stop** - Disables new user connections to a database and unloads every element of the database from the memory of their respective hosts. This option does the same thing as if you perform a close and unload operation of the database.

- **Close** - Disables new user connections to a database.

- **Unload** - Unloads every element of the database from the memory of their respective hosts.

Figure 2–88   Select an option
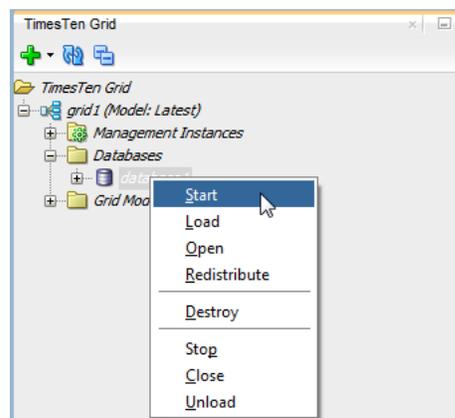
SQL Developer performs the operation that you selected. The operation can take a few minutes.

> **Note:**   If you select the stop, close, or unload option, a warning dialog displays asking you to confirm that you want to continue with the operation. The dialog contains a number hyperlink that you can click to view the current connections to your database. If you are sure that you want to proceed with the operation, click **Yes**.

A progress dialog displays in the lower right corner of the main SQL Developer window. If SQL Developer displays an error message, troubleshoot the error.

You can review the status of your database by double clicking the name of your database from the TimesTen Grid view region. If your database status is already open, you can click the refresh button that is located in the top left corner of the database status window to refresh your database status.

You have successfully performed an operation on the database.

Working with connectables

SQL Developer enables you to perform various operations with direct and client/server connectables. A connectable creates a DSN in the configuration files of every data instance. Applications may use this DSN to connect to the database. The connection attributes found in the DSN depend on the type of connectable. For more information about connection attributes, see "Connection Attributes" in the *Oracle TimesTen In-Memory Database Reference*.

There are two types of available connectables:

- **Direct connectable:** Creates a system DSN with the connection attributes both defined in the database definition and the connectable.

- **Client/server connectable:** Creates a client DSN with the `TTC_SERVER` attribute of one or all the data instances available plus the connection attributes defined in the connectable.

You can perform these operations with connectables:

- Create a connectable

- Edit a connectable

- Export a connectable

- Delete a connectable

Create a connectable

SQL Developer enables you to create client/server and direct connectables.

To create a client/server or direct connectable, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Expand the **Databases** option, expand the **name of your database**, then either right-click the **Client/Server Connectables** or **Direct Connectables** option and select **New Connectable**.

Figure 2–89   Select New Connectable
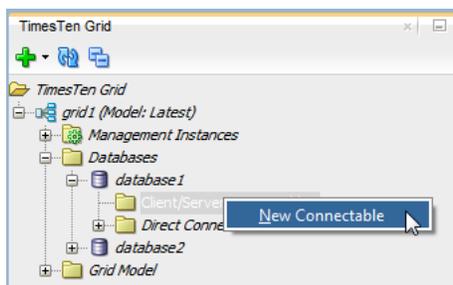
The New Connectable dialog displays. Locate the Connectable Name text field.

2. In the **Connectable Name** text field, type a name for your connectable. Select a name that is a meaningful identifier.

3. From the **Connect to instance** drop-down list, select a specific instance that applications connect to through the client/server DSN. The default value is Any Instance which lets your application connect to your database through any available data instance.

> **Note:** Optionally, you can import a connectable file to load connection attribute values for your connectable. The connectable file must use .connect as the file name extension.
>
> To import a connectable file, click the **Import Definition** button. Then, select **your connectable file** and click **Open**. If a connection attribute value was set in the wizard before you imported the connectable file, the connection attribute value changes to the value that is defined in the connectable file.
>
> Once you have imported the connectable file, you can still edit the connectable definition within the wizard. If you do not want to keep editing the connectable definition, skip to step 5.
>
> For more information on the connectable file, see "Creating a connectable file" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

4. Set the connection attributes for your connectable.

   You can add additional connection attributes by selecting a connection attribute from the drop-down list located below the Connect to instance drop-down list and then clicking **Add**. If a connection attribute is not listed in the drop-down list, you can select the **New Attribute** option and enter the connection attribute name in the Attribute column of the connection attribute table. Depending on the connection attribute, you can edit the value with a text field or a drop-down list of valid values.

   You can remove a connection attribute by selecting a connection attribute from the connection attribute table and then clicking **Delete**.

   For more information on valid connection attributes and values, see "Connection Attributes" in the *Oracle TimesTen In-Memory Database Reference*.

5. Once you have set your connection attributes, click **OK**.

Figure 2–90   Click OK
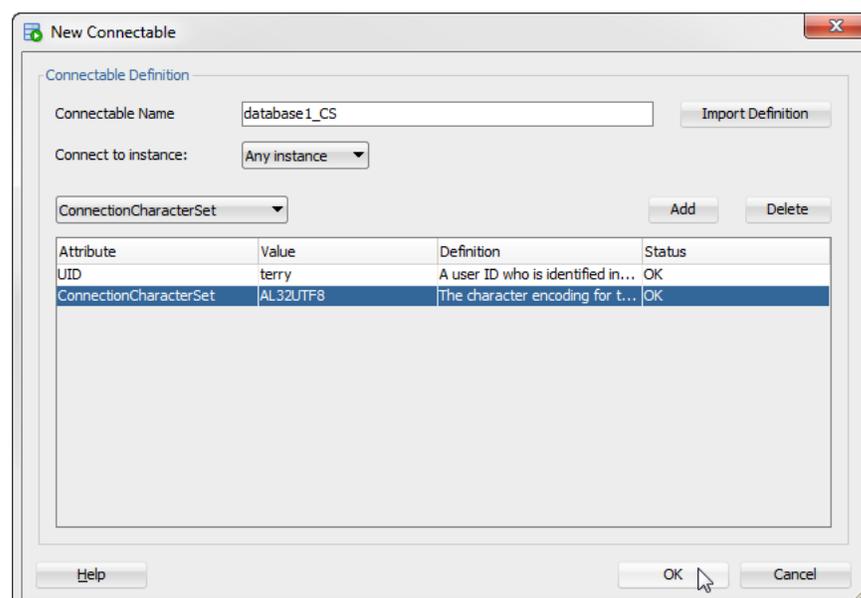
The New Connectable dialog closes and the Creating connectable progress dialog displays. Creating a connectable can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button.

6. Click **Close**.

The Creating connectable progress dialog closes.

You have successfully created a connectable for your database. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

Edit a connectable

SQL Developer enables you to edit an existing connectable.

To edit a connectable, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

Expand the **Grid Model** option, expand the **Database Definitions** option, expand the **name of your database**, then either expand the **Client/Server Connectables** or **Direct Connectables** option, right-click the **name of the connectable** and select **Edit**.

*Figure 2–91   Select Edit*
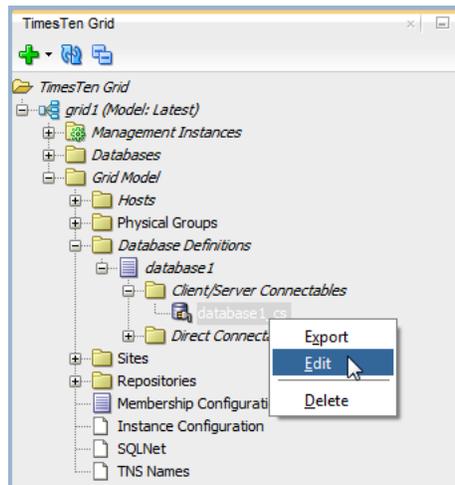
The Edit Connectable dialog displays.

Follow the steps, starting on step 3, as described in "Create a connectable" on page 2-90.

Export a connectable

SQL Developer enables you to export an existing connectable. You can use this exported connectable file to import the definition of a new connectable.

To export a connectable, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

Expand the **Grid Model** option, expand the **Database Definitions** option, expand the **name of your database**, then either expand the **Client/Server Connectables** or **Direct Connectables** option, right-click the **name of the connectable** and select **Export**.

Figure 2–92   Select Export
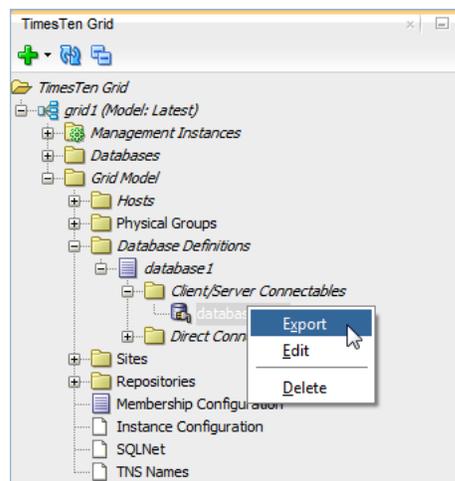
The Save Connectable definition dialog displays.

Specify the location of where you want to export the connectable file.

In the **File Name** text field, specify a meaningful name for the **connectable file** and click **Save**.

The Save Connectable Definition dialog closes.

You have successfully exported a connectable file.

Delete a connectable

SQL Developer enables you to delete a connectable from your grid model.

To delete a connectable, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

Expand the **Grid Model** option, expand the **Database Definitions** option, expand the **name of your database**, then either expand the **Client/Server Connectables** or **Direct Connectables** option, right-click the **name of the connectable** and select **Delete**.

Figure 2–93   Select Delete
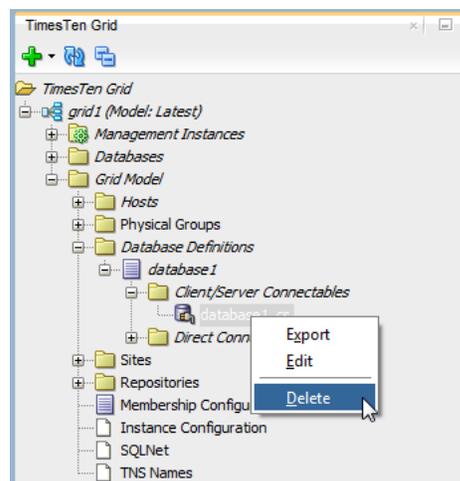
The Delete Connectable dialog displays. Locate the Yes button.

Click **Yes**.

The Delete Connectable dialog closes and the Delete Connectable progress dialog displays. Locate the Close button.

Click **Close**.

You have successfully deleted a connectable. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

View active connections to the database

SQL Developer enables you to view all active connections to a database. This is useful to determine if it is safe to perform certain database operations like destroying a database or closing a database.

To view active connections to a database, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

Expand the **Databases** option and select **the name of your database**.

Figure 2–94   Select the name of your database
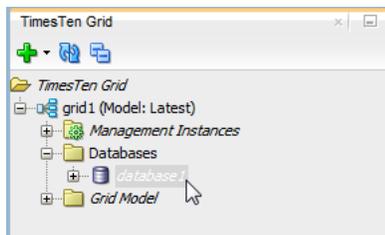
The database status displays or refreshes if it is already displayed. Locate the connections link that is next to the Number of applications connected to the *database_name* label.

Click the **connections link** that is next to the Number of applications connected to the *database_name* label.

Figure 2–95   Click the connections link
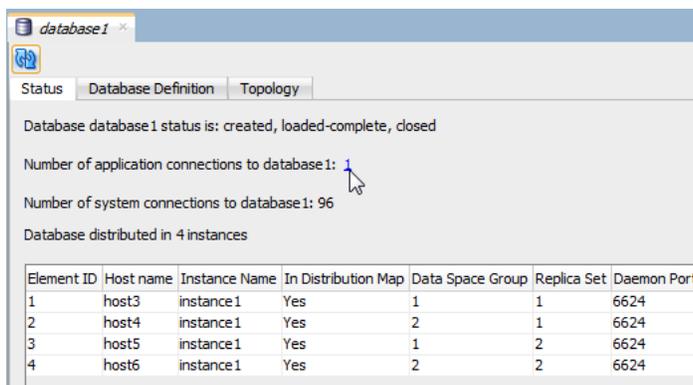
The Updating information for database progress dialog displays. Once the database information has been loaded, the Updating information for database progress dialog closes and the Connection Details dialog displays.

Figure 2–96   The Connections Details dialog
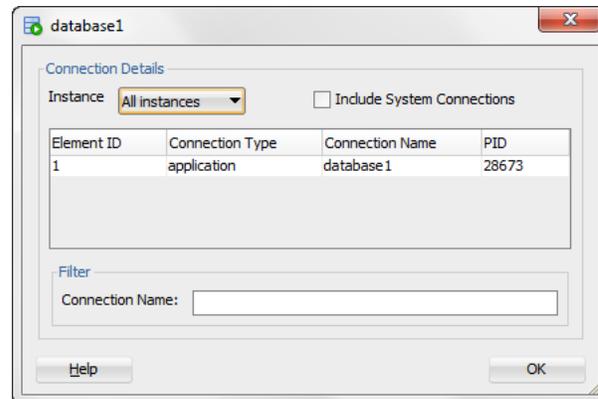
The Connection Details dialog enables you to view all connections to your database data instances or specific data instances. Use the Instance drop-down list to select an instance for which you want to see its connection. In case that you want to see system connections, select the Include System Connections check box. You can also use the Connection Name text field to create a filter for all available connections to your selected instance.

Click the **OK** button to close the Connection Details dialog.

Destroy a database

SQL Developer enables you to destroy a database.

Before you create a grid:

- Ensure that you have migrated or backed up all data from the database. For more information, see "Migrating, Backing Up and Restoring Data" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

- Ensure that the database is unloaded from memory. See "Perform database operations" on page 2-88 for more information.

To destroy a database, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Expand the **Databases** option, then right-click the **name of your database** and select **Destroy**.

Figure 2–97   Select Destroy
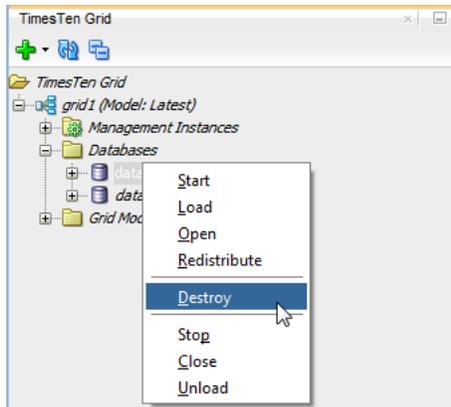
The Destroy database dialog displays.

Ensure that there are no active connections to your database before proceeding. Click the number hyperlink near the top of the dialog to view the connection details of your database. Ensure that there are no critical connections to your database or properly close any active connections.

If there are 0 active connections to the database, you are ready to proceed.

2. In the text field, enter the **instance administrator name**.

3. If you want to delete the database definition of your database, click in the check box to the left of the text **Delete database definition**.

4. Click **OK**.

Figure 2–98   Click OK
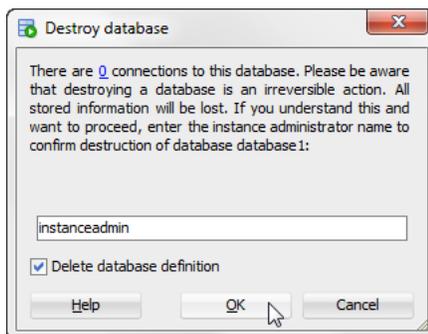
The Destroying database dialog displays. Deleting a database can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

The Locate the Close button.

5. Click **Close**.

The Destroying database progress and Destroy database dialogs close.

You have successfully deleted a database. However, you need to apply your changes to the grid. Changes made to the grid model do not immediately impact a grid until you explicitly apply these changes. See "Apply grid model changes to the grid" on page 2-33 for more information on applying model changes to a grid.

Working with repositories

In a grid, a repository is used to store backups of databases, database exports, and collections of log files. TimesTen Scaleout enables you to define a repository as a directory path mounted using NFS on each host or as a directory path that is not directly mounted on each host. Multiple grids can use a single repository as long as each grid is associated with that repository.

SQL Developer enables you to create and detach repositories from your grid. For more information about repositories, see "Working with repositories" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

SQL Developer enables you to perform the following procedures with repositories:

- Create a repository
- Detach a repository

Create a repository

To create a new grid repository, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Expand the **Grid Model** option, right-click **Repositories** and select **Create Repository**.

Figure 2–99 Select Create Repository
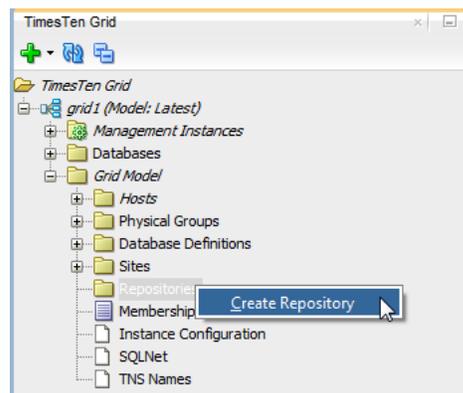
The Create Repository dialog displays. Locate the Repository Name field.

2. In the **Repository Name** field, specify the name for the repository.

3. In the **Location** field, specify a location that is mounted using NFS on each host or as a directory path that is accessible on each host with SSH or SCP.

4. From the **Method** drop-down list, select a mounting option:

   - Mount - Creates a repository as a directory path mounted using NFS on each host of your grid.

   - SCP - Creates a repository as a directory path that is not directly mounted on each host of your grid.

5. If you selected the SCP method, specify the fully qualified domain name of the host on which the repository exists in the **Address** field.

6. Click **OK**.

Figure 2–100   Click OK
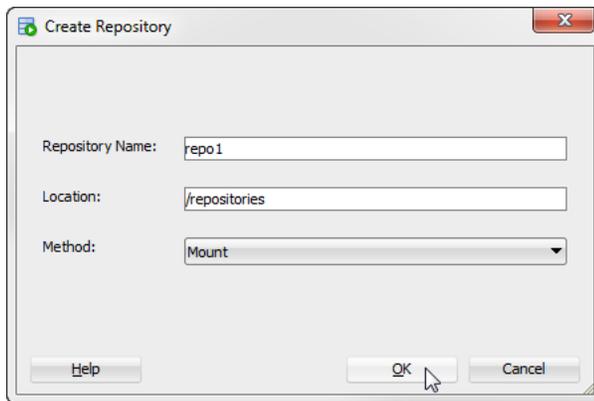
The Create Repository dialog closes and the Create Repository progress dialog displays. Creating the repository can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button in the Create Repository progress dialog.

7. Click **Close**.

The Create Repository progress dialog closes.

You have successfully created a repository. This repository is now available in the repositories folder of your grid model.

Detach a repository

To detach a grid repository, ensure that you are on the main SQL Developer page, that you have enabled the TimesTen Grid view, and that your grid node is expanded.

1. Expand the **Grid Model** option, expand **Repositories**, then right-click the **name of your repository** and select **Detach**.

Figure 2–101   Select Detach
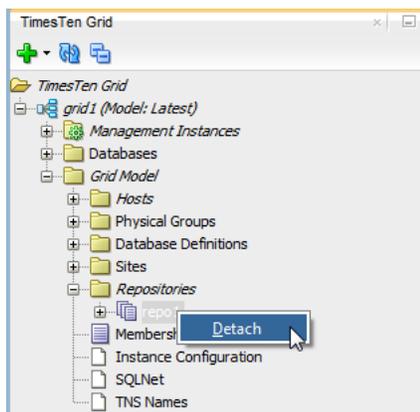
The Detach Repository warning dialog displays. Locate the Yes button.

2. Click **Yes**.

The Detach Repository warning dialog closes and the Detach Repository progress dialog displays.Detaching the repository can take a few minutes. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button in the Detach Repository progress dialog.

3. Click **Close**.

    The Detach Repository progress dialog closes.

You have successfully detached a repository.

Inspecting logs

SQL Developer enables you to review the support and user log of TimesTen Scaleout. These logs may be useful in troubleshooting errors with your grid. For each instance that is part of your grid, you can view these logs:

■   Support Log

■   User Error Log

To view logs, ensure that you are on the main SQL Developer page and that you have enabled the TimesTen Grid view.

1. Right-click your grid node, select **Inspect Logs,** then select **Support Log** or **User Error Log**.

Figure 2–102   Select an option
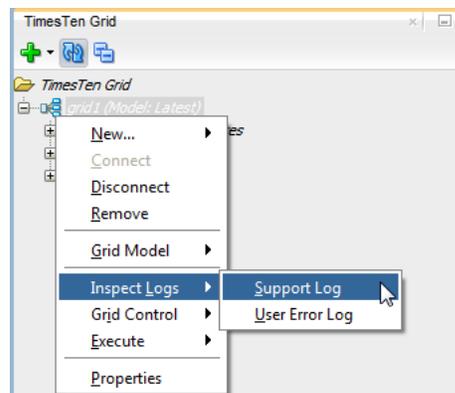
Depending on the option that you select, the Inspect Support Log or Inspect User Error log dialog displays.

2. In the **Available instance** folder, locate the instance for which you want to view a log. Right-click the **instance name** and select **Load/Unload**. If you want to view a log of additional instances, right-click the corresponding instance name and select Load/Unload for any additional instances.

> **Note:**   In case the log panel expands and hides the instance panel of the dialog, place your cursor at the left edge of the log panel and use the resize arrow to resize the log panel.

If you want to close a log of an instance, locate the instance in the Showing Logs folder for which you want to close a log. Then, right-click the **instance name** and select **Load/Unload.**

If you want to create filters, save the filtered logs, refresh logs, or perform various other operations with the logs, click the **Logs** menu button. The Logs menu button

is located in the top left corner of the Inspect Support Log or Inspect User Error log dialog.

3. After you finish reviewing the log, click **Cancel**.

   Depending on the option that you select, the Inspect Support Log or Inspect User Error log dialog closes.

You have successfully reviewed the support log or user error log of the instances of your grid.

3

Connecting to a TimesTen Database

You can connect to a TimesTen database after you install and start SQL Developer.

This chapter focuses on the steps necessary to connect to a TimesTen database in SQL Developer. The topics include:

- Defining a TimesTen database named connection
- Using an existing TimesTen database named connection
- Disconnecting an existing TimesTen database connection

Defining a TimesTen database named connection

To create a named connection to a TimesTen database, ensure that you are on the main SQL Developer page.

1. Right-click the **Connections** node and select **New Connection**.

Figure 3–1 Creating a new connection
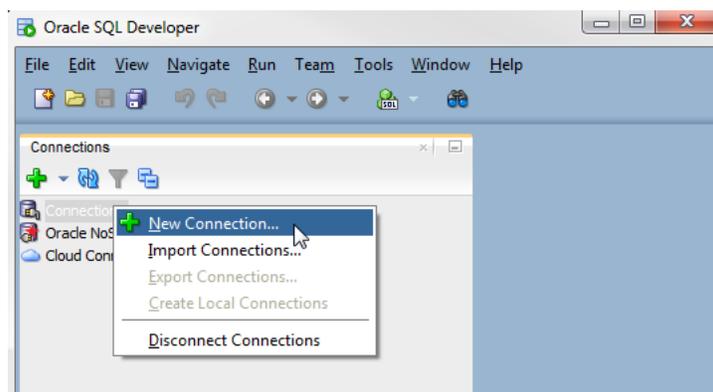
The **New/Select Database Connection** dialog appears.

2. Select the **TimesTen** tab.

Figure 3–2   New database connection
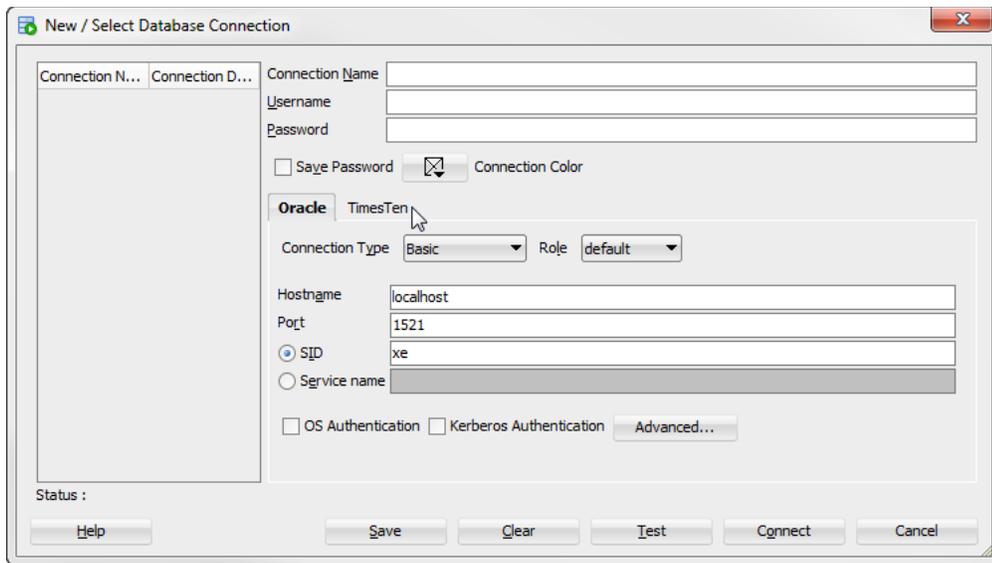
You are ready to enter information for the connection.

3. In the **Connection Name** field, enter the connection name.

4. In the **Username** field, enter the TimesTen user. If you use the TimesTen database to cache data from an Oracle database, enter the TimesTen cache manager user name. If you use the TimesTen database to load data from an Oracle database, enter the TimesTen user name that has SELECT privileges on the Oracle database tables that you want to load.

5. In the **Password** field, enter the password for the TimesTen user.

   To save the password, choose **Save Password**. If you save the password, you will not be prompted for the password on subsequent connections to the TimesTen database.

6. To allow TimesTen to issue an implicit commit after DML statements, select the **Autocommit** check box. By default, the **Autocommit** check box is selected.

7. Select a connection type from the **Connection Type** drop-down list. The following types of connections are available:

   > **Note:** Caching data from an Oracle database is not supported by TimesTen Scaleout databases.

   - DSN
   - Easy Client/Server
   - Advanced

Figure 3–3   New connection type
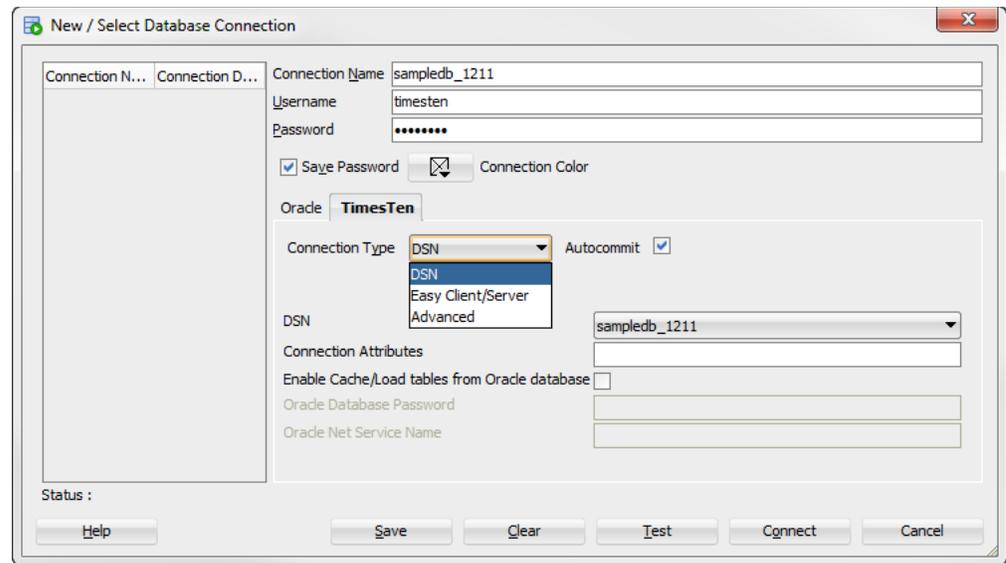
You are now ready to define your new connection.

DSN

The DSN connection type allows you to connect to databases by specifying a DSN. The DSN must exist in your `.odbc.ini` on Linux or UNIX and the ODBC Data Source Administrator on Windows. Before defining a DSN connection in SQL Developer, ensure that you have completed the steps up to step 7 from "Defining a TimesTen database named connection" on page 3-1.

> **Note:**   For a TimesTen Scaleout database, use the `ttGridAdmin gridClientExport` command to export every client DSN available for the grid into a flat file. Then, import the client DSNs into the TimesTen client system where SQL Developer is installed. For more information, see "Establishing client connections from a TimesTen Client" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

> **Note:**   For more information about defining DSNs for TimesTen databases, see "Specifying Data Source Names to identify TimesTen databases" in the *Oracle TimesTen In-Memory Database Operations Guide*.

1. Select a predefined DSN that references the TimesTen database from the **DSN** drop-down list. You can select either a Data Manager DSN to establish a direct connection or a client DSN to establish a client/server connection.

2. In the **Connection Attributes** field, enter the attributes for the DSN. The **Connection Attributes** field is not required. Each attribute setting must be separated by a semicolon. If you do not specify any connection attributes, TimesTen uses the connection attributes specified in the DSN.

    For more information about connection attributes, see "Connection Attributes" in the *Oracle TimesTen In-Memory Database Reference*.

3. To cache data from an Oracle database or load data from an Oracle database into a TimesTen database, select the **Enable Cache/Load tables from Oracle database** check box.

4. In the **Oracle Database Password** field, enter the Oracle database password for the Oracle database user. The Oracle database user must be the same user as the TimesTen user you specified in the **Username** field. This text field is only available if you selected the **Enable Cache/Load tables from Oracle database** check box.

5. In the **Oracle Net Service Name** field, enter the Oracle database net service name for the Oracle database. This text field is only available if you selected the **Enable Cache/Load tables from Oracle database** check box.

6. Click **Save** to save the connection name under the **Connections** node.

Figure 3–4   Save DSN connection
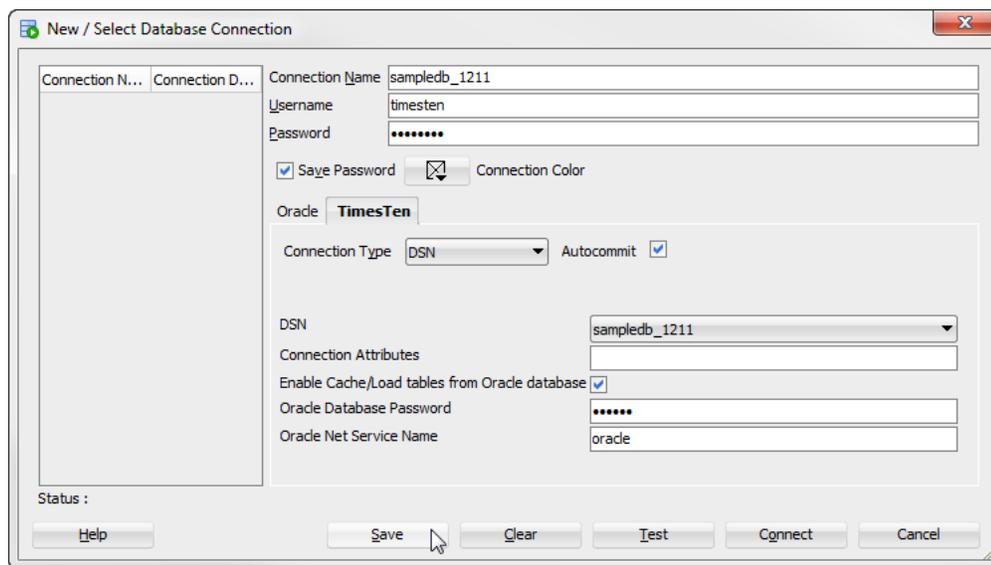
Your connection is saved. You can establish connections to the database using the settings in the named connection without having to specify the values in the fields of the **New/Select Database Connection** dialog box again.

7. Click **Connect**.

SQL Developer is connected to your TimesTen database.

For more information about creating and managing database connections in SQL Developer, see the *Oracle SQL Developer User's Guide*.

Easy Client/Server

The easy client/server connection type allows you to connect to local and remote TimesTen databases without having to define a client DSN. Before defining an easy client/server connection in SQL Developer, ensure that you have completed the steps up to step 7 from "Defining a TimesTen database named connection" on page 3-1.

> **Note:**   For a TimesTen Scaleout database, the Easy Client/Server connection enables you to connect to a single data instance. If the data instance is unreachable the connection fails, which makes the Easy/Client connection type suitable only for testing purposes.

1. In the **Server Host Name** field, enter the name or the IP address of the TimesTen server host.

2. In the **Server DSN** field, enter the DSN of the TimesTen database that is located on the TimesTen server host.

3. In the **Server Port** field, enter the TCP port of the TimesTen server. You can use the `ttStatus` utility to find the TCP port of your TimesTen server.

   In the following example, the server port number is 53396. This example is part of a sample output from the `ttStatus` utility.

   ```
   TimesTen server pid 24379 started on port 53396
   ```

   For more information about the `ttStatus` utility, see "ttStatus" in the *Oracle TimesTen In-Memory Database Reference*.

4. In the **Client Timeout(s)** field, enter the `TTC_TIMEOUT` connection attribute. Setting this attribute is optional. A value of 0 indicates that client/server operations should not timeout. If this attribute is not set, the default timeout period is 60 seconds. For more information, see "TTC_Timeout" in the *Oracle TimesTen In-Memory Database Reference*.

5. To cache data from an Oracle database or load data from an Oracle database into a TimesTen database, select the **Enable Cache/Load tables from Oracle database** check box.

6. In the **Oracle Database Password** field, enter the Oracle database password for the Oracle database user. The Oracle database user must be the same user as the TimesTen user you specified in the **Username** field. This text field is only available if you selected the **Enable Cache/Load tables from Oracle database** check box.

7. In the **Oracle Net Service Name** field, enter the Oracle database net service name for the Oracle database. This text field is only available if you selected the **Enable Cache/Load tables from Oracle database** check box.

8. Click **Save** to save the connection name under the **Connections** node.

Figure 3–5   Save easy client/server connection

Your connection is saved. You can establish connections to the database using the settings in the named connection without having to specify the values in the fields of the **New/Select Database Connection** dialog box again.

9. Click **Connect**.

   SQL Developer is connected to your TimesTen database.

   For more information about creating and managing database connections in SQL Developer, see the *Oracle SQL Developer User's Guide*.

Advanced

The advanced connection type allows you to define a connection string to connect to your TimesTen database. Before defining an advanced connection in SQL Developer, ensure that you have completed the steps up to step 7 from "Defining a TimesTen database named connection" on page 3-1.

1. Select a driver type for the TimesTen database from the **Driver Type** option. Select either **Client/server** to establish a client/server connection or **Direct** to establish a direct connection.

2. In the **Connection string** field, enter a connection string to define the DSN and set attributes for the DSN. If you choose Client/Server as the Driver Type specify the server host, server port, and DSN. Each attribute setting must be separated by a semicolon. Do not delimit the connection string with a pair of double quotation marks.

   For a TimesTen Scaleout database, ensure that you specify all of the TTC_SERVER*n* connection attributes for your database connectable. The TTC_SERVER*n* is the network address and TCP port number of the data instances on which your database is distributed. Use the ttGridAdmin gridClientExport command to view all of the connectables of your TimesTen Scaleout databases. For example, a valid connection string for a TimesTen Scaleout database might look like:

   ```
   TTC_SERVER1=host1.example.com/6625;TTC_SERVER2=host2.example.com/6625;TTC_SERVE
   R3=host3.example.com/6625;TTC_SERVER4=host4.example.com/6625;TTC_SERVER_DSN=dat
   abase1;UID=timesten;
   ```

   For information on the ttGridAdmin utility and the format and contents of a connection string, see "ttGridAdmin" in the *Oracle TimesTen In-Memory Database Reference*. and "Connecting to a database using a connection string" in the *Oracle TimesTen In-Memory Database Operations Guide*, respectively.

3. To cache data from an Oracle database or load data from an Oracle database into a TimesTen database, select the **Enable Cache/Load tables from Oracle database** check box.

4. In the **Oracle Database Password** field, enter the Oracle database password for the Oracle database user. The Oracle database user must be the same user as the TimesTen user you specified in the **Username** field. This text field is only available if you selected the **Enable Cache/Load tables from Oracle database** check box.

5. In the **Oracle Net Service Name** field, enter the Oracle database net service name for the Oracle database. This text field is only available if you selected the **Enable Cache/Load tables from Oracle database** check box.

6. Click **Save** to save the connection name under the **Connections** node.

Figure 3–6   Save advanced connection
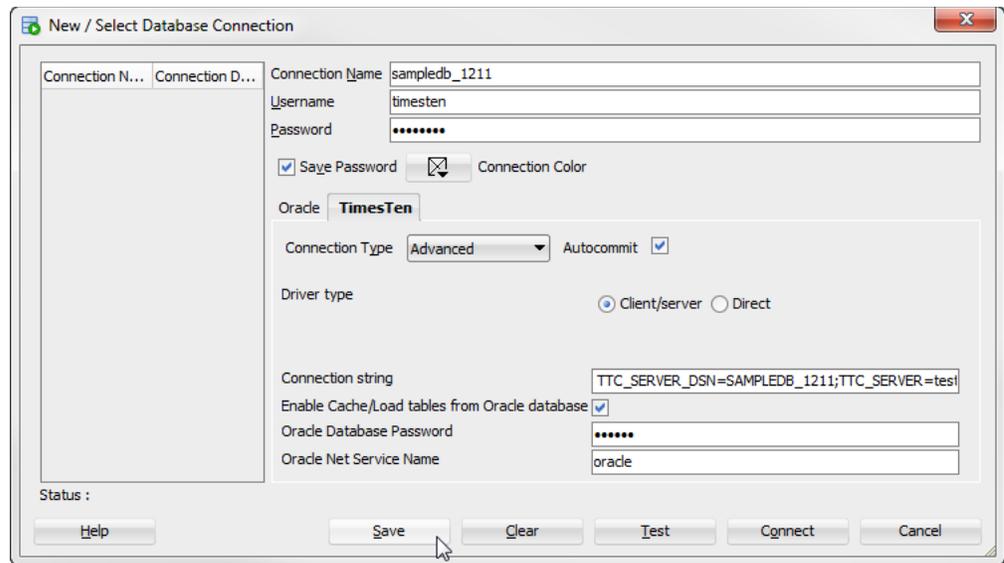
Your connection is saved. You can establish connections to the database using the settings in the named connection without having to specify the values in the fields of the **New/Select Database Connection** dialog box again.

7.   Click **Connect**.

SQL Developer is connected to your TimesTen database.

For more information about creating and managing database connections in SQL Developer, see the *Oracle SQL Developer User's Guide*.

Using an existing TimesTen database named connection

If a named connection for the TimesTen database exists under the **Connections** node, then right-click the node of the connection name and select **Connect**, or click the **+** to the left of the connection name's node.

Figure 3–7   Viewing a TimesTen connection
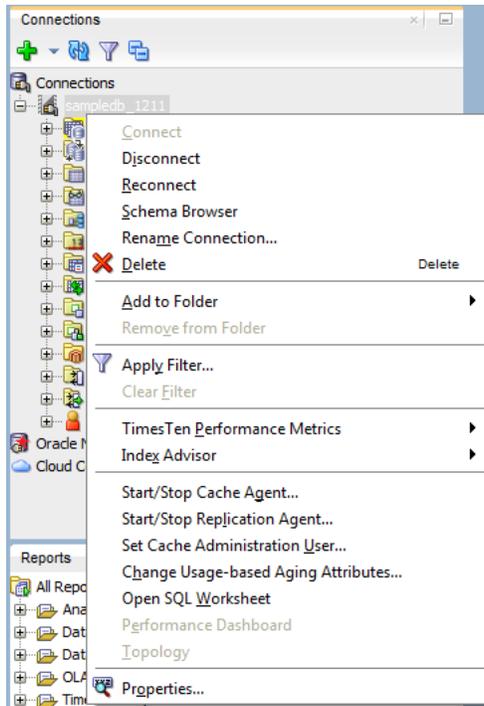
If the **Connection Information** dialog appears, in the **Username** field, enter the TimesTen user name. In the **Password** field, enter the password for the user.

Click **OK**.

Figure 3–8   Connection Information dialog
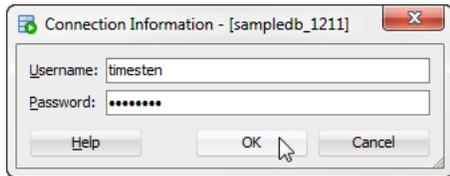
After you connect to a TimesTen database, you can browse existing objects or create new objects such as tables, indexes, views, sequences, synonyms, PL/SQL packages, procedures and functions and so on. For information about creating database objects, see the *Oracle SQL Developer User's Guide*.

Disconnecting an existing TimesTen database connection

To disconnect an existing connection from a TimesTen database, choose the **Connections** tab. Right-click the node of the connection name and select **Disconnect**.

For more information, see Figure 3–7, "Viewing a TimesTen connection".

4

Working with TimesTen Objects

This chapter focuses on working with TimesTen objects in SQL Developer.

Topics include:

- Working with tables
- Working with PL/SQL
- Working with sequences
- Working with views
- Working with materialized views
- Locating a TimesTen database object
- Granting and revoking object privileges
- Viewing the privileges granted on a database object
- ttIsql commands in SQL worksheet

Working with tables

SQL Developer enables you to work with tables for the TimesTen databases. This section includes these topics:

- Create a table
- Edit a table
- Review the distribution of data for a TimesTen Scaleout table
- Create a primary key constraint
- Change the attributes of primary key
- Create an index on a table
- Create a foreign key constraint
- Specify and display LOB data types
- Specify the INLINE attribute for columns
- Alter the aging properties of a table
- Load data into tables
- Compute table size information
- View the characteristics of a table

■   [Drop a constraint](#)

Create a table

You can create a table for the TimesTen database or TimesTen Scaleout database with SQL Developer.

To create a table, ensure that you are on the main SQL Developer page and that your TimesTen or TimesTen Scaleout connection is expanded.

1.   Right-click the **Tables** option and select **New Table**.

Figure 4–1   Select New Table
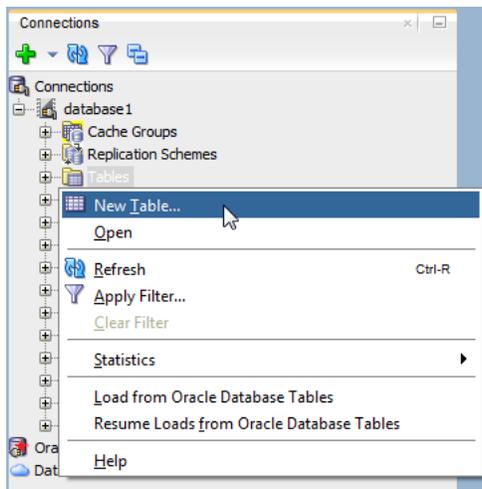
The Create Table dialog displays. Locate the Schema drop-down list.

2.   From the **Schema** drop-down list, select the owner of the table.

3.   In the **Name** text field, type a name for the table.

You are now ready to define the columns for your table. Locate the Columns region.

Define the columns of your table

To define the columns of your table, locate the Columns table. For each of the columns that you want to define:

1.   In the **PK** column, click in the column if you want to define this column as a primary key. You need to have a primary key if you want to create an index on your table.

2.   In the **Name** column, type a meaningful column name. For more information on supported column names, see "Basic names" in the *Oracle TimesTen In-Memory Database SQL Reference*.

3.   In the **Data Type** column, select a data type for your column. For more information on data types, see "Data Types" in the *Oracle TimesTen In-Memory Database SQL Reference*.

4.   In the **Size** column, select a size for your column based on your the data type that you selected.

5.   If you want this column to be NOT NULL, select the **Not Null** check box.

6. If you want this column to have a default value, type a value in the **Default** column.

7. If you want to define this column as INLINE, select the **Inline** check box. By default, variable-length columns whose declared column length is > 128 bytes are stored out of line. Variable-length columns whose declared column length is <= 128 bytes are stored inline.

You have successfully defined a column for your table.

To add another column, click the **+** button that is located in the top right corner of the Columns table. Then, repeat steps 1-7 to define each column.

To remove a column, select the column that you want to remove and click the **-** sign that is located in the top right corner of the Columns table.

To copy a column, select the column that you want to copy and click the **Copy Column** button that is located in the top right corner of the Columns table.

Figure 4–2 Define the columns for your table
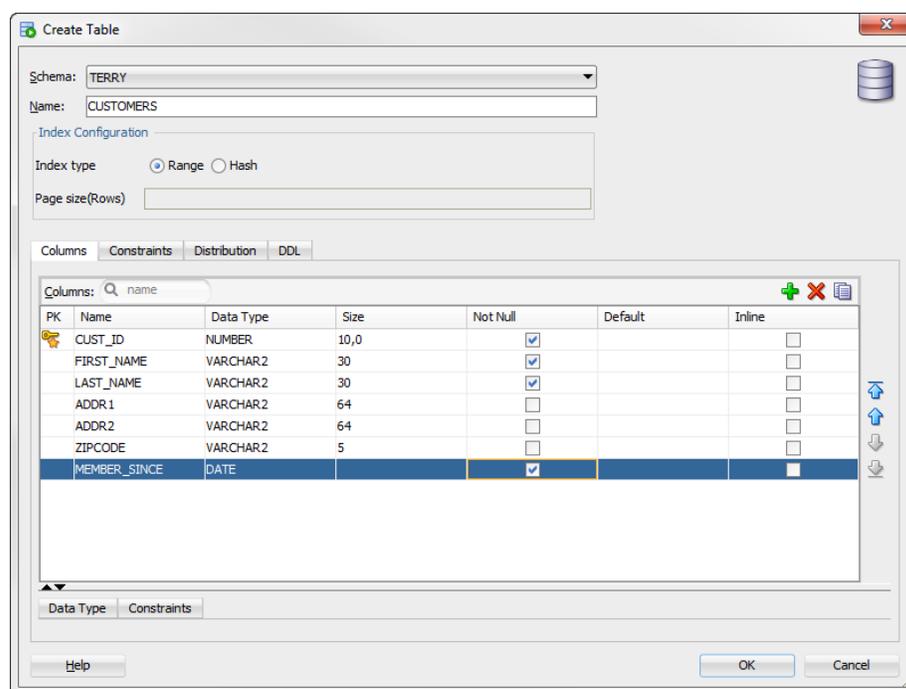
Once you have defined all of the columns for your table:

■ If you defined a primary key for your table, create an index for your table. See "Create an index for your table" on page 4-3 for more information.

■ If you did not define a primary key on your table, set constraints for your table. See "Set constraints for your table" on page 4-4 for more information.

Create an index for your table

To define an index for your table, you need to have defined a primary key for your table. To define an index on your table:

Select an index type from the **Index Type** options. For more information on index types, see "CREATE TABLE" in the *Oracle TimesTen In-Memory Database SQL Reference*.

If you selected a hash index, type a `RowPages` parameter in the **Page size(Rows)** text field.

Compute this value by dividing the number of expected rows in your table by 256. For example, if your table has 256,000 rows, specify 1000 for the value of `RowPages` (256000/256=1000).

Figure 4–3   Create an index for your table
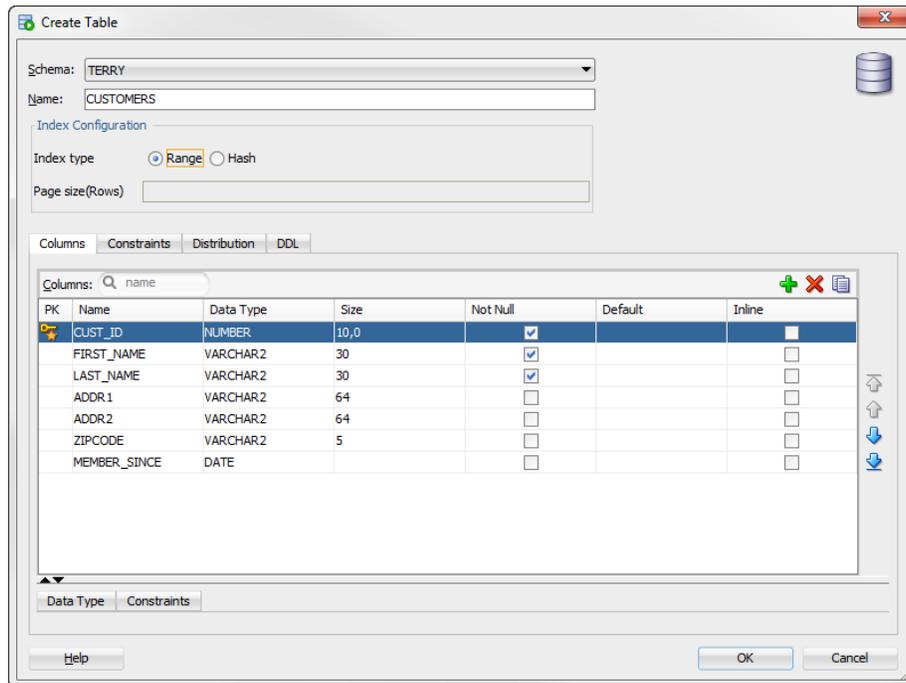
You have successfully defined an index on your table. You are now ready to define constraints for your table.

Set constraints for your table

Select the **Constraints** tab to work with constraints for your table.

If you have defined a primary key or compound primary key in the columns tab, you see this primary key constraint in the Constraints tab.

Figure 4–4   Set the constraints for your table
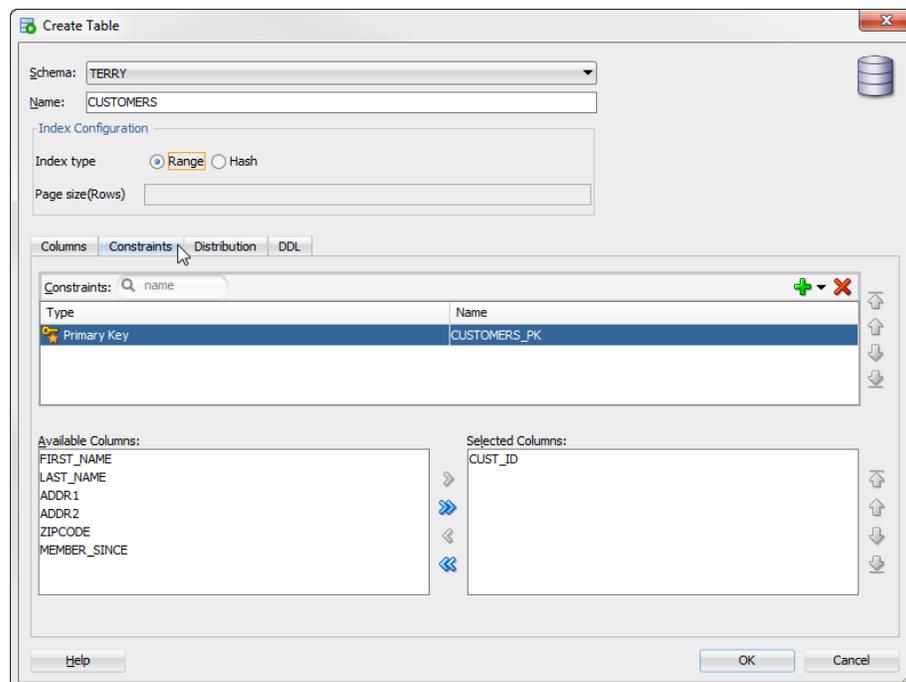
You can set these types of constraints:

■   Primary key constraint

■   Foreign key constraint

Primary key constraint If you have not defined a primary key for your table in the Columns tab, you can click the **+** button located in the top right corner of the Constraints region and then select **New Primary Key Constraint**.

If you have already defined a primary key, the New Primary Key Constraint option is not available. However, you are able to edit the definition of your primary key in the constraints region.

In the Constraints table, ensure that the name of your primary key is a meaningful identifier. You can edit the name of a primary key by editing the name column of the Primary Key row in the Constraints table.

To select columns to be part of your primary key definition, double click the column name in the **Available Columns** list.

To remove columns from your primary key definition, double click the column name in the **Selected Columns** list.

To remove a primary key constraint, select the primary key constraint that you want to remove and click the **x** sign that is located in the top right corner of the Constraints region.

Foreign key constraint To define a foreign key for your table:

1. Click the + button located in the top right corner of the Constraints region and then select **New Foreign Key Constraint**.

2. In the Constraints table, ensure that the name of your foreign key is a meaningful identifier. You can edit the name of a foreign key by editing the name column of the Foreign Key row in the Constraints table.

Locate the Referenced Constraint region.

3. From the **Schema** drop-down list, select the schema name of the table that you want to reference. Alternatively, you can type the schema name instead of selecting it from the drop-down list.

4. From the **Table** drop-down list, select the table name that you want to reference. Alternatively, you can type the table name instead of selecting it from the drop-down list.

5. From the **Constraint** drop-down list, select a primary key constraint name that you want to reference. Alternatively, you can type the primary key constraint name instead of selecting it from the drop-down list.

6. From the **On Delete** drop-down list, select an on delete option.

You have successfully defined a foreign key constraint.

To add another foreign key constraint, the + button located in the top right corner of the Constraints region and then select **New Foreign Key Constraint**. Then, repeat steps 1-6 to define each foreign key constraint.

To remove a foreign key constraint, select the foreign key constraint that you want to remove and click the **x** sign that is located in the top right corner of the Constraints region.

Set the distribution method for your table

This tab is only available when working with a TimesTen Scaleout table.

To define the distribution method for your table:

1. Select the **Distribution** tab.

2. From the **Distribution method** drop-down list, select a distribution method for your table:

    ■   Distribute by Hash

        Distribution scheme that distributes data based on the hash of the primary key or a set of user-specified columns. For more information, see "Hash" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

    ■   Distribute by Reference

        Distribution scheme that distributes the data of a child table based on the location of the corresponding parent row of a foreign key constraint. Your table needs at least one foreign key to be able to use this distribution scheme. For more information, see "Reference" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

    ■   Duplicate

        Duplicate is a distribution scheme that distributes full identical copies of the table's data to all the elements of a database. For more information, see "Duplicate" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

Distribute by Hash When you decide how to distribute your table with the hash distribution scheme, consider the following:

■   If the table has a primary key and you do not specify a distribution, column, TimesTen Scaleout distributes the data based on the hash of the primary key.

- If the table does not have a primary key or you do not specify a distribution column, TimesTen Scaleout distributes the data based on the hash of a hidden column that TimesTen Scaleout adds for this purpose.

- If you specify a distribution column, TimesTen Scaleout distributes the data based on the hash of the distribution column. You can specify multiple distribution columns.

To specify distribution columns for your table:

To select columns to be part of your distribution column, double click the column name in the **Available Columns** list.

To remove columns from your distribution column, double click the column name in the **Selected Columns** list.

You are now ready to review the DDL of your CREATE TABLE statement. See "Review the DDL of the CREATE TABLE statement" on page 4-7 for more information.

Distribute by Reference Ensure that you have defined a foreign key to use the distribute by reference distribution scheme. To distribute your table with this distribution scheme:

From the **Foreign Key** drop-down list, select the foreign key to distribute the data of your table.

You are now ready to review the DDL of your CREATE TABLE statement. See "Review the DDL of the CREATE TABLE statement" on page 4-7 for more information.

Duplicate When you select the distribute by duplicate distribution scheme, you do not need to specify any additional fields in the Distribution tab of the Create Table dialog.

You are now ready to review the DDL of your CREATE TABLE statement. See "Review the DDL of the CREATE TABLE statement" on page 4-7 for more information.

Review the DDL of the CREATE TABLE statement

Select the **DDL** tab to review the DDL of your CREATE TABLE statement.

> **Note:** To save the DDL statement in a .sql file, click the **Save** button.

Once you have reviewed the DDL statement, click **OK**.

Figure 4–5   Click OK
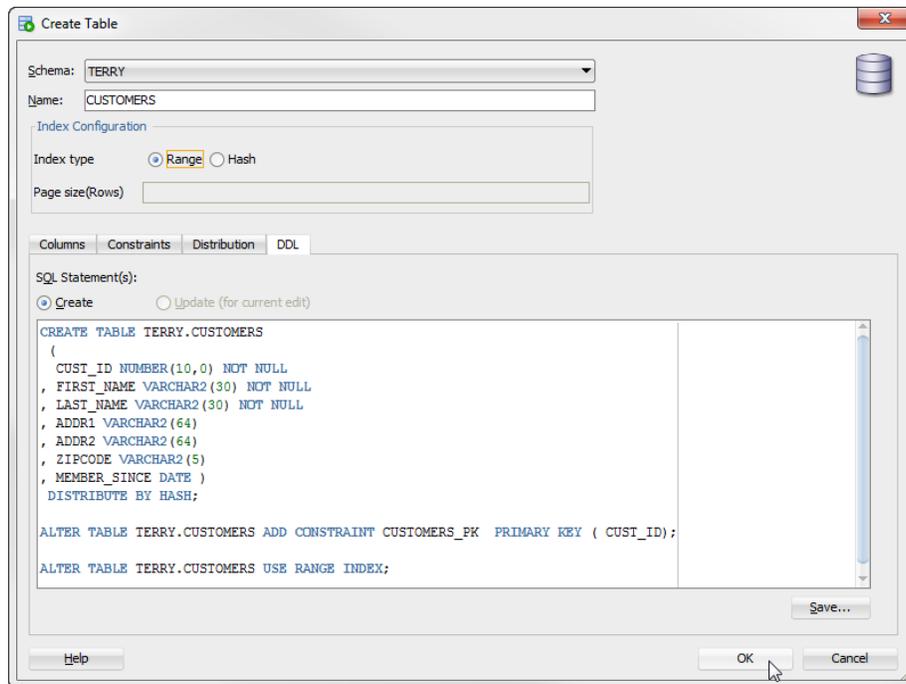
The Create Table dialog closes and TimesTen Scaleout creates your table.

You have successfully created a table.

Edit a table

SQL Developer enables you to alter a table that is already created. When you alter an existing table, you cannot perform these actions:

- Change the schema name.

- Change the table name.

- Change the index.

- Change the distribution scheme of a table that contains data. This only applies to TimesTen Scaleout tables.

To edit a table, ensure that you are on the main SQL Developer page and that your connection is expanded:

Expand the **Tables** option, right-click the **table name**, and then select **Edit**.

Figure 4–6   Select Edit
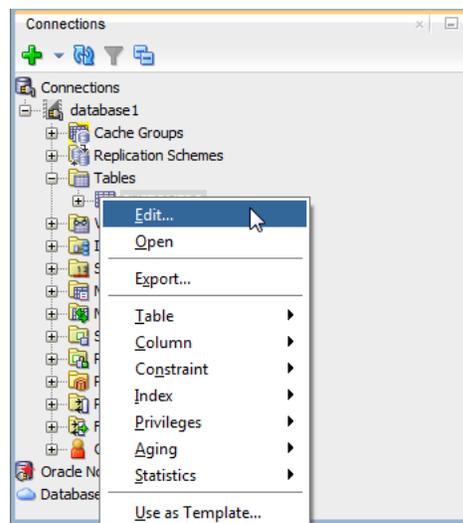
The Edit Table dialog displays. The Edit Table dialog is the same as the Create Table dialog, except that not all options can be edited. You can edit all of the options except for those that are mentioned above. See "Create a table" on page 4-2 for more information on how to use the Create Table dialog.

Review the distribution of data for a TimesTen Scaleout table

SQL Developer enables you to view distribution statistics for your table. This feature is only available for TimesTen Scaleout tables. Distribution statistics enable you to see how your data is distributed between the data instances of your grid.

To view the distribution statics for a table, ensure that you are on the main SQL Developer page and that your connection is expanded:

Expand the **Tables** option and double-click the **table name**.

A table view displays. Locate the Distribution tab.

Click the **Distribution** tab from the table view.

Figure 4–7   Click the Distribution tab

The Distribution tab displays.

Figure 4–8   Distribution tab
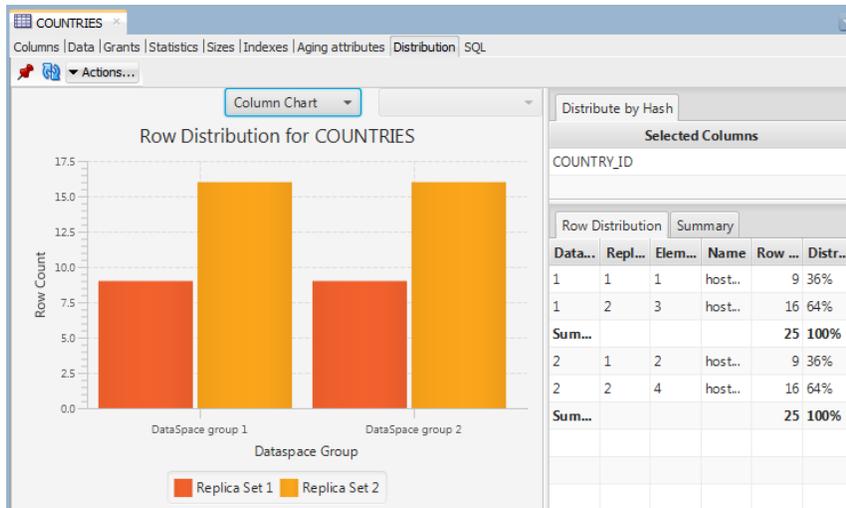
The Distribution tab contains these regions:

> **Note:**   The contents of each region can vary depending on the distribution scheme of the table.

- Row distribution chart

  This chart can show either a column or pie chart of the row count of your table on each of your elements. Use the drop-down list in this region to switch between the column and pie chart for your table.

  The pie chart enables you to view a pie chart for the row count of each element of your data space groups. Use the drop-down list in this region to switch between data space groups. The pie chart is not available when your table uses a duplicate distribution scheme.

  If one of your elements is down or unavailable, the corresponding column chart for that element becomes unavailable. Also, the pie chart for the data space group of the element becomes unavailable.

- Distribution scheme table

  This table shows you information about the distribution scheme that your table uses.

  - Distribute by hash distribution scheme - This table shows the columns that TimesTen Scaleout uses to create a hash in order to distribute the rows of your table.

  - Distribute by reference distribution scheme - This table shows the foreign key relationship that TimesTen Scaleout uses to distribute the rows of your table.

  - Duplicate distribution scheme - This table doesn't show any information because this distribution scheme has the same information on every element.

- Row distribution table

  This table shows the row count of your table on each of your elements. You can also see information about the data space group, replica set, element ID, instance

name, and distribution percentage for each of the elements. If you use the duplicate distribution scheme, the distribution percentage column is not available.

If one of your elements is down or unavailable, the row count and distribution percentage columns for that element are marked as unavailable.

If the table is distributed by hash or reference, the distribution tab will display a summary row including totals for row count and distribution. This summary is only accessible when the table is sorted by data space group and will be hidden if sorted on a different column. If needed, this information is always available in the "Summary" tab next to "Row Distribution".

If the table is distributed by hash or reference, this table displays summary rows with total row count and distribution percentage for each data space group. However, this is only available if the row distribution table is sorted by data space groups. If you use a different sort order, you can click the Summary tab to see these summary rows.

- Summary table

  This table shows the row count and distribution percentage for each data space group.

Create a primary key constraint

You can create a primary key constraint on a table that does not have a primary key defined. Follow these steps:

1. Click the **+** to the left of the **Tables** node to view the list of tables. Then right-click the name of the table to create a primary key constraint.

2. Select **Constraint**, then select **Add Primary Key**.

    The **Add primary key** dialog appears.

3. In the **Key index type** field choose either **Use range** to create a primary key using a range index or **Use hash** to create a primary key using a hash index.

    If you choose **Use hash,** you see the text **with number of** followed by a drop-down list of either **row** or **pages**. Select either **row** or **pages** and in the box after the **colon** (**:**), enter the number of rows or pages. The default selection is **row**.

    For more information on hash index sizing, see "CREATE INDEX" in the *Oracle TimesTen In-Memory Database SQL Reference*.

4. In the **Constraint Name** field, enter the constraint name. The name cannot exceed 30 characters in length.

5. In the **Column** field(s), you see a drop-down list of column names. Choose the column name(s) to use for the primary key.

6. Click **Apply**.

    The **Confirm** dialog box appears indicating that the primary key constraint has been added.

7. Click **OK**.

    The primary key constraint has been added successfully.

To view the SQL statement used to create the primary key constraint, choose the **SQL** tab of the **Add primary key** dialog box.

For information about creating a primary key constraint, see "CREATE TABLE" in the *Oracle TimesTen In-Memory Database SQL Reference* or "ALTER TABLE" in the *Oracle TimesTen In-Memory Database SQL Reference*.

Change the attributes of primary key

You can change the attributes of a primary key constraint after you have defined a primary key on a table. Follow these steps:

1. Click the **+** to the left of the **Tables** node to view the list of tables. Then right-click the name of the table on which to change the primary key attributes.

2. Select **Constraint**, then select **Change Primary Key Parameters**.

   The **Change parameters of a primary key** dialog appears.

3. In the **Key index type** field choose either **Use range** to change the primary key constraint from using a hash index to using a range index or **Use hash** to change the primary key constraint from using a range index to using a hash index.

   If you choose **Use hash,** you see the text **with number of** followed by a drop-down list of either **row** or **pages**. Select either **row** or **pages** and in the box after the **colon** (**:**), enter the number of rows or pages. The default selection is **row**.

   For more information on hash index sizing, see "CREATE INDEX" in *Oracle TimesTen In-Memory Database SQL Reference*.

4. Click **Apply**.

   The **Confirm** dialog appears indicating that the index(es) for the primary key constraint has been changed.

5. Click **OK**.

   The index(es) for the primary key constraint has been changed successfully.

To view the SQL statement used to change the attributes of the primary key constraint, choose the **SQL** tab of the **Change parameters of a primary key** dialog.

For information about changing the attributes of a primary key constraint, see "ALTER TABLE" in the *Oracle TimesTen In-Memory Database SQL Reference*.

Create an index on a table

You can create a range index or a hash index on a regular table or on a cache table.

To create an index, click the **+** to the left of the **Tables** node to view the list of tables. Then right-click the name of the table to create an index on and select **Index**, then select **Create Index**.

Figure 4–9   Creating an index from Tables node
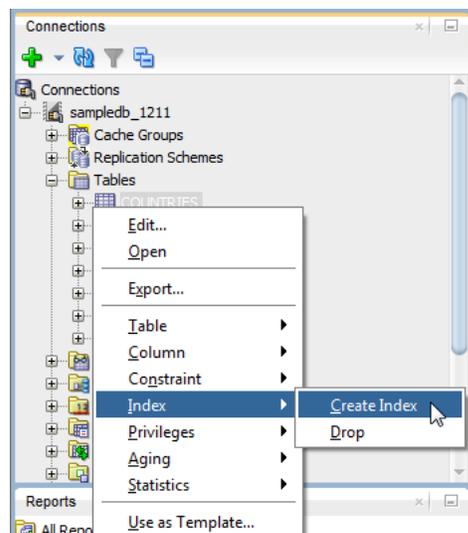
You can also right-click the **Indexes** node and select **New Index**.

Figure 4–10   Creating a new index from Indexes node
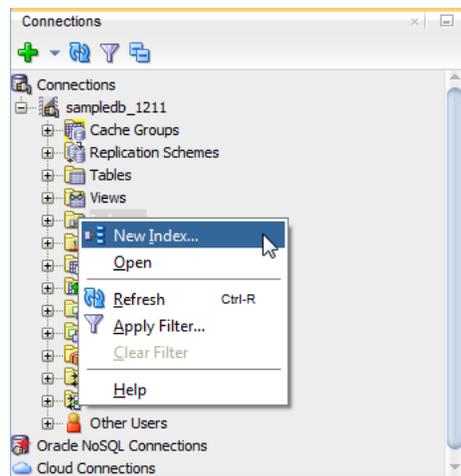
In the **Properties** tab of the **Create Index** dialog, locate the **Table** drop-down menu. From this menu, select the table for the index.

For **Index Type**:

■   To create a range index, choose **Range**.

■   To create a hash index, choose **Hash**. Then, select a **Pages** value:

–   **Page size(Rows)** - Specify the number of pages to use. To determine the value, divide the number of expected rows in your table by 256. For example, if your table has 256,000 rows, specify 1000 (256000/256=1000).

–   **Current** - The current number of rows in the table is used to calculate the page count value. Do not use this option if there are no rows in your table. This is because when rows are added to the table, the hash index performs poorly.

For **Uniqueness**:

- To have a single occurrence of index key column values in your table, choose **Unique**. If not, select **Non-unique**.

For information about the different types of indexes supported in TimesTen, see "CREATE INDEX" in the *Oracle TimesTen In-Memory Database SQL Reference*.

In the **Index** section, specify the columns for the index. You can select the columns from the **Column Name** and then click **>** to add these columns to your index. For each indexed column, from the **Order** select list, you can specify whether the column is to be sorted in ascending or descending order. The default sort order is ascending.

To create a composite index, select additional columns and then click **>**. Select a column and then the **<** to remove columns from the index definition.

Figure 4–11   Create index dialog
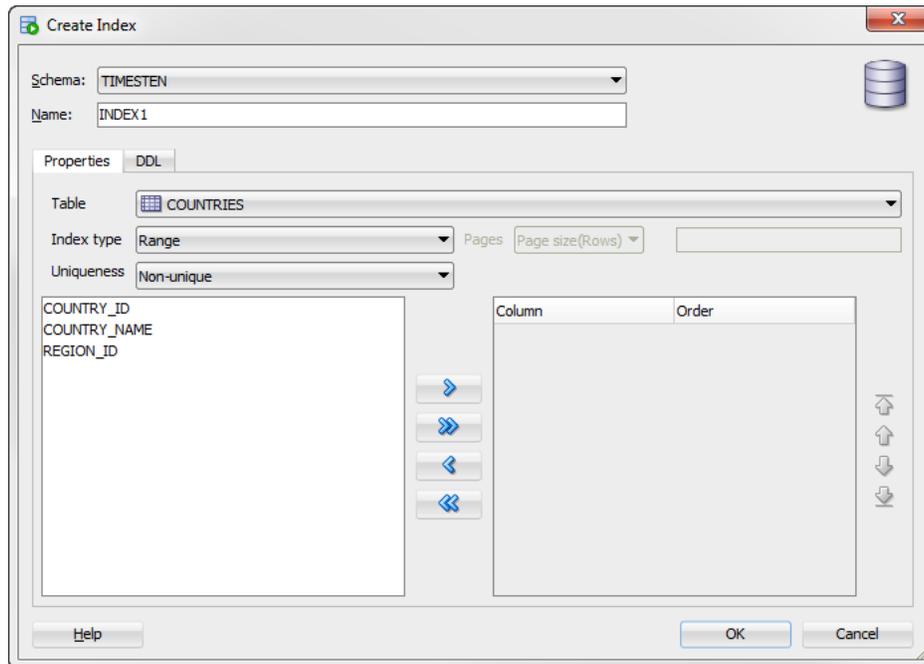
In the **DDL** tab of the **Create Index** dialog, you can view the `CREATE INDEX` statement used to create the index.

Click **OK** to create the index.

Figure 4–12   Viewing the DDL for create index
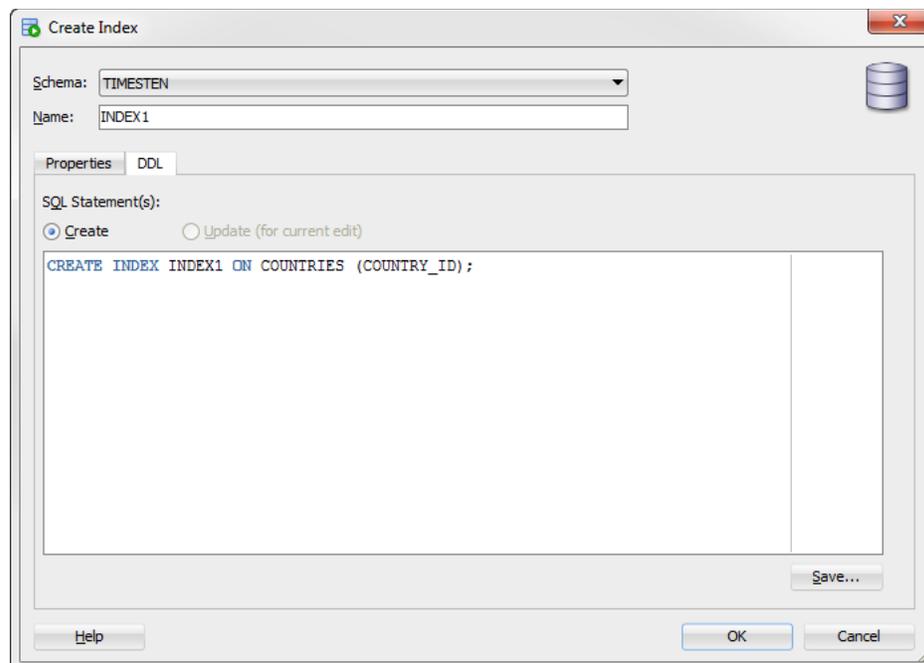
Create a foreign key constraint

You can create a foreign key constraint on a table. Ensure that you are on the main SQL Developer page and that your TimesTen connection is expanded.

1. Click the **+** to the left of the **Tables** node to expand the list of tables.

Figure 4–13   Expand Tables
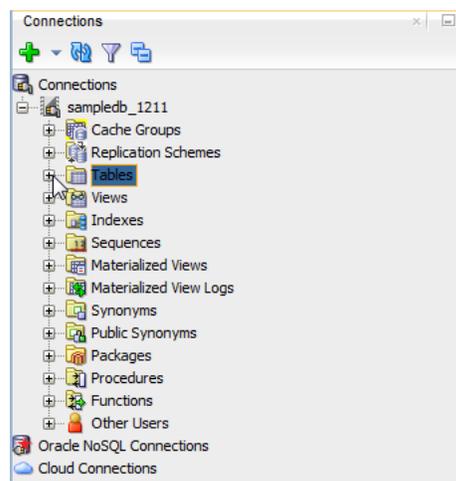
2. Right-click the name of the table and select **Constraint**, then select **Add Foreign Key**.

Figure 4–14   Add Foreign Key
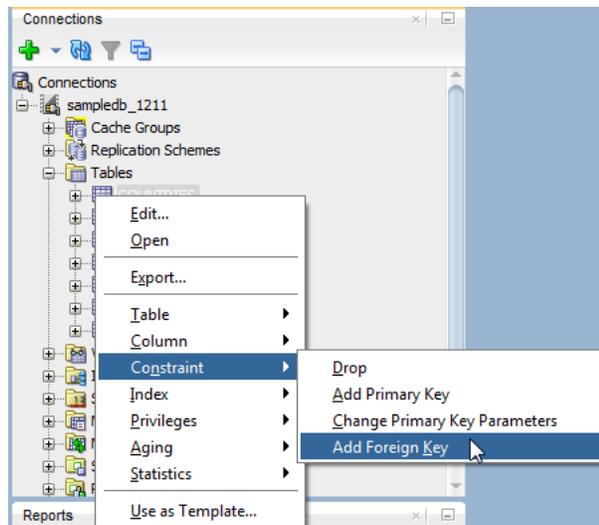
The **Add Foreign Key** dialog displays. The Owner and Name fields are auto-filled and you cannot edit these fields. Prepare to enter the Constraint Name.

3. To enable the ON DELETE CASCADE referential action, select the **Cascade delete foreign key** checkbox. If specified, when rows containing referenced key values are deleted from a parent table, rows in child tables with dependent foreign key values are also deleted.

4. In the **Constraint Name** field, enter the name of the foreign key.

5. Select the column on which to apply the foreign key constraint from the **Column Name** drop-down list.

6. Select the table name that the foreign key references from the **Referenced Table Name** drop-down list.

7. Select the column that the foreign key references from the **Referenced Column** drop-down list.

    To view the SQL statement that will be used to create the foreign key constraint, choose the **SQL** tab of the **Add Foreign Key** dialog box.

8. Click **Apply**.

Figure 4–15   Click Apply
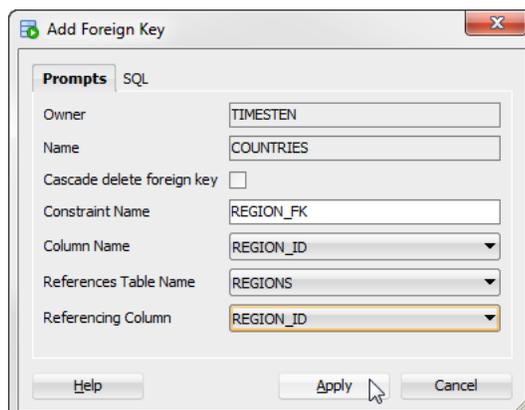
The **Confirmation** dialog box displays indicating that the foreign key constraint has been added.

9. Click **OK**.

Figure 4–16   Click OK

The foreign key constraint has been added successfully.

For information about creating a foreign key constraint, see "CREATE TABLE" and "ALTER TABLE" in the *Oracle TimesTen In-Memory Database SQL Reference*.

Specify and display LOB data types

If you are using TimesTen release 11.2.2 or later, then you can specify and display LOB data types in your table definitions. If you are using TimesTen Scaleout, ensure that this feature is supported in your version of TimesTen Scaleout. See "Comparison between TimesTen Scaleout and TimesTen Classic" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

If you want to create a table with a LOB data type, then in the **Create Table** dialog, expand the **Type** column header. You see CLOB, NCLOB, and BLOB as valid data types.

Figure 4–17   Specifying LOB data types in the Create Table dialog
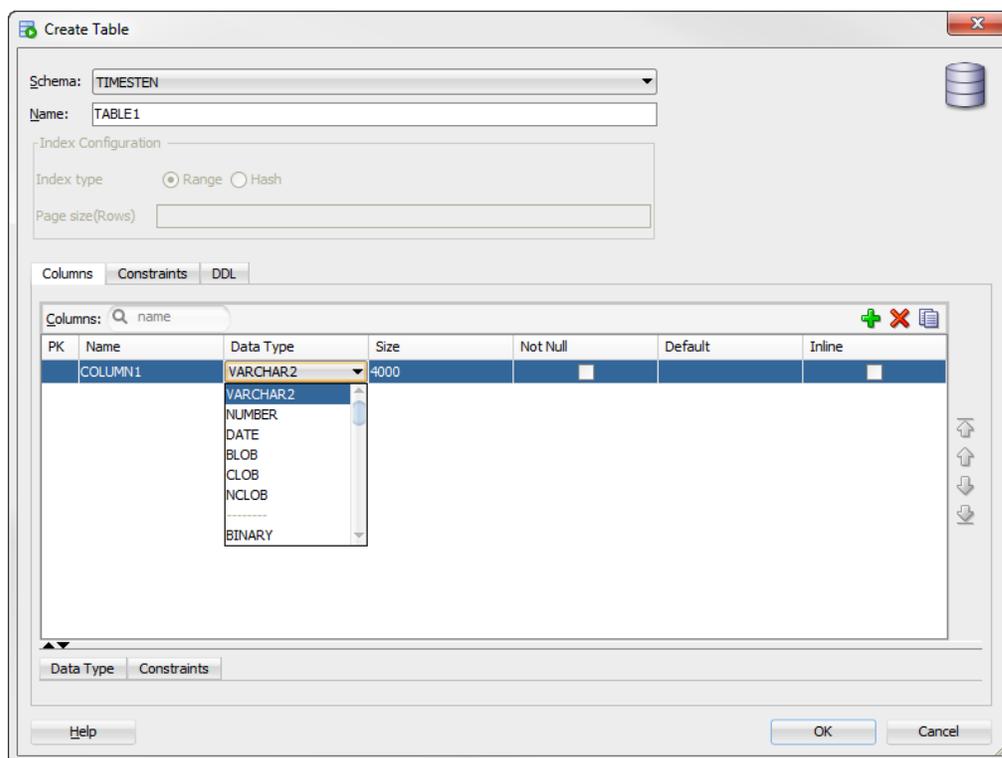
To view the data type for your columns, select the table and then choose the **Columns** tab. You see the LOB data type for your column.

Figure 4–18   Displaying Column data types
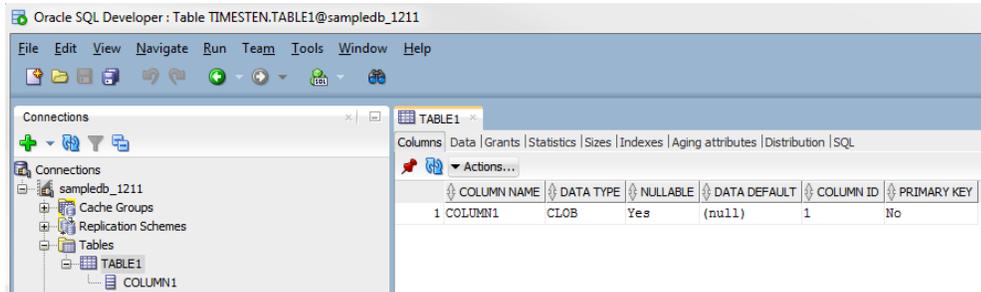
If you want to add a column and specify a LOB data type, use the Add Column option. Right-click on the table, select **Column**, then select **Add**. The **Add Column** dialog appears. In the **Add Column** dialog, expand **Data Type**. You see CLOB, NCLOB, and BLOB among the possible choices.

Figure 4–19   Adding column and specifying a LOB data type
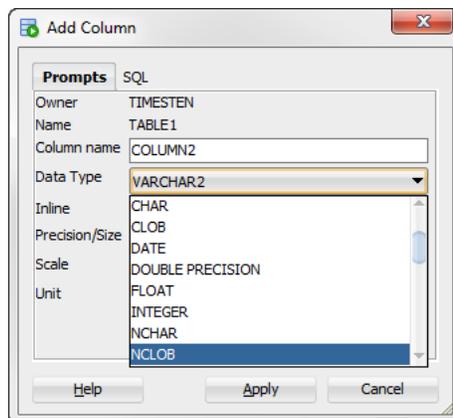
For PL/SQL objects, you can specify LOB data types as IN, OUT, and IN OUT parameters. In the PL/SQL edit dialogs, for parameter types, choose CLOB, NCLOB, or BLOB.

Figure 4–20   Choosing a parameter type
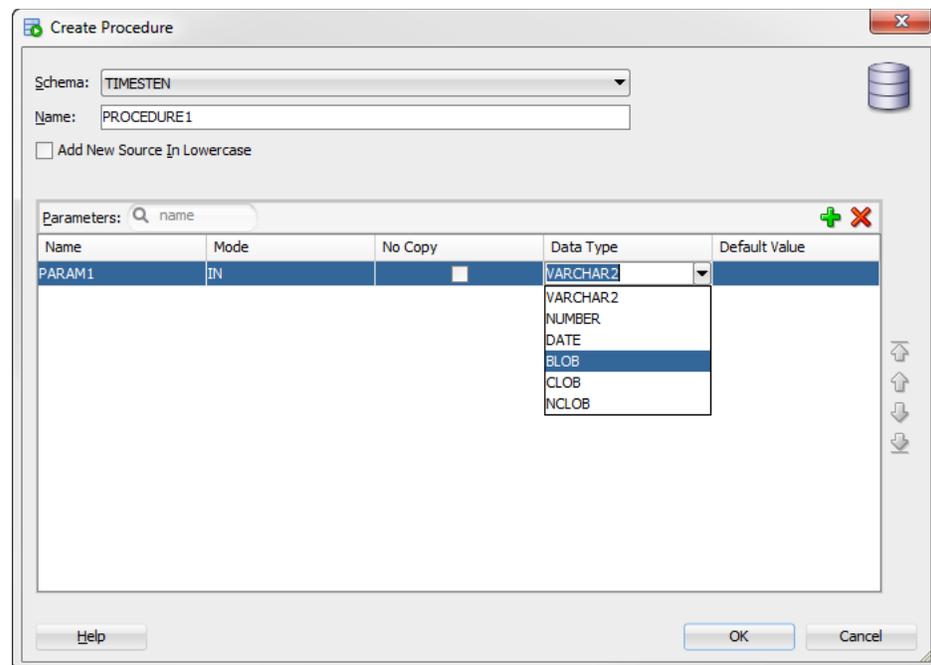
Specify the INLINE attribute for columns

You can specify the INLINE attribute for columns of type VARCHAR2, NVARCHAR2, and VARBINARY.

In the **Create Table** dialog, locate the column header named **Inline**. Click in the check box to define the column with the INLINE attribute.

*Figure 4–21   Viewing the Inline attribute in the Create Table dialog*
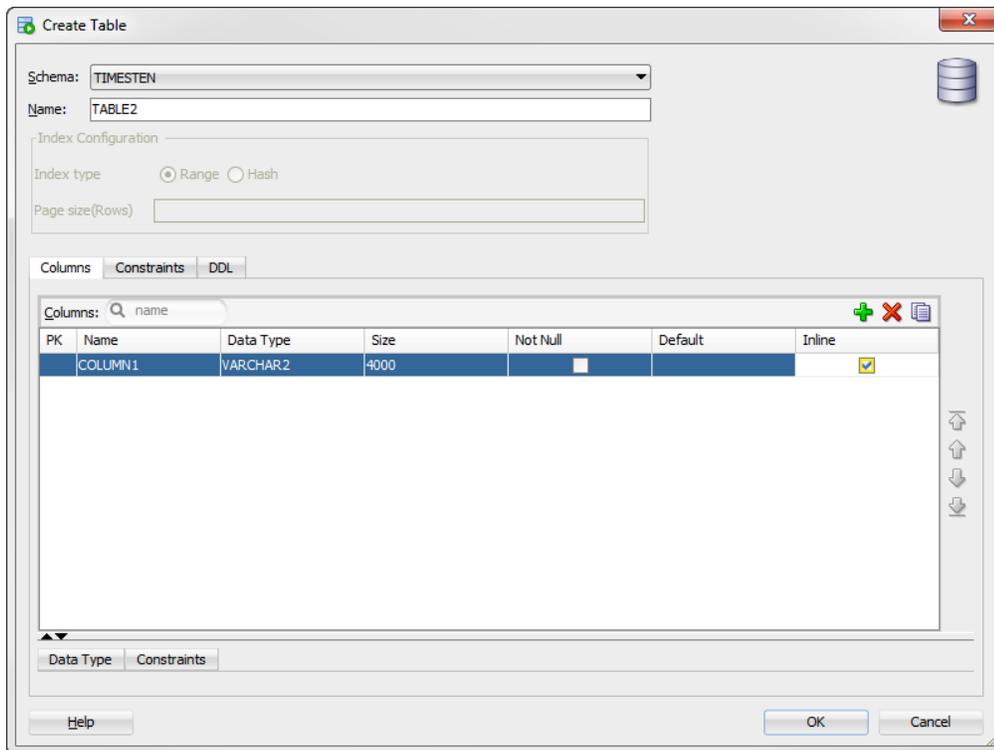
You can also specify the INLINE attribute when adding a column to a table. Right-click on the table, select **Column**, then select **Add**. The **Add Column** dialog appears. For columns of type VARCHAR2, NVARCHAR2, and VARBINARY, click in the **Inline** check box to add the column with the INLINE attribute.

Figure 4–22   Viewing the Inline attribute in the Add Column dialog
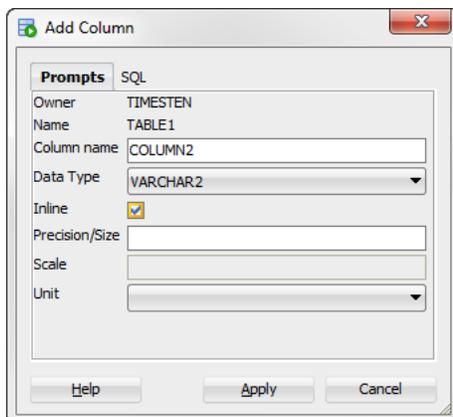
Alter the aging properties of a table

You can add an aging policy to or drop an aging policy from a regular table or a cache table. You can also change the aging state for a table's existing aging policy, or change the lifetime and cycle for a table's existing time-based aging policy. An aging policy is only supported on TimesTen tables.

Adding an aging policy to a table

To add an aging policy to a regular table or a cache table, under the **Tables** node, right-click the name of the table to add an aging policy to and select **Aging**, then select **Add Usage-based** to add an LRU aging policy. To add a time-based aging policy select **Aging**, then select **Add Time-based**. An aging policy can be added to a cache table only if it is the root table of a cache group.

Figure 4–23   Aging attributes
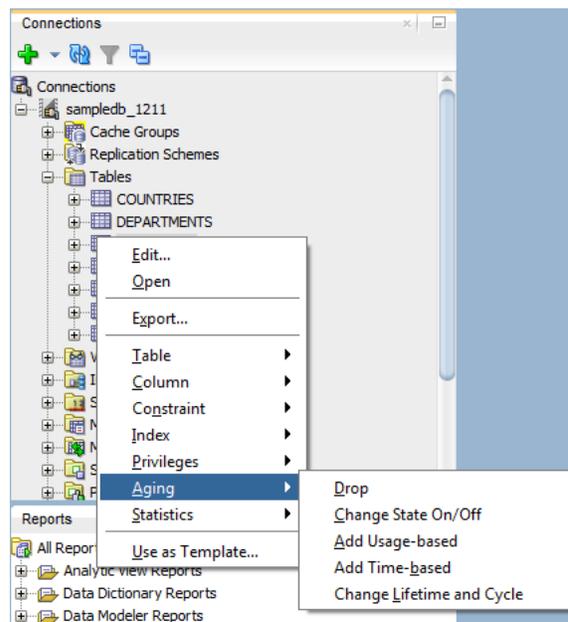
If you are unable to find the table that you are looking for, see "Locating a TimesTen database object" on page 4-52.

To add an LRU aging policy to the table, in the **Prompts** tab of the **Add Usage-based** dialog box, specify an aging state by choosing either **On** or **Off** in the **Usage-based aging** field. The default aging state is on. Click **Apply** to add the LRU aging policy to the table.

Figure 4–24   Adding usage-based aging
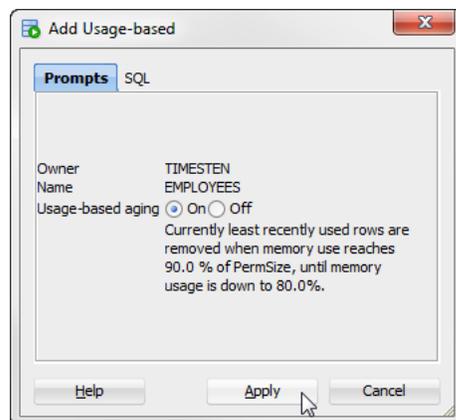
The TimesTen database memory usage thresholds determine when data starts and stops being deleted from the table. The default memory usage threshold that

determines when data starts being deleted from the table is 90%. The default memory usage threshold that determines when data stops being deleted from the table is 80%. The default LRU aging cycle is 1 minute. For information about how to change these thresholds and the LRU aging cycle., see "Specifying an aging policy" on page 9-8.

An LRU aging policy can be added to any regular table, and only to cache tables in an AWT, SWT or user managed cache group that does not have automatic refresh defined.

To add a time-based aging policy to the table, in the **Prompts** tab of the **Add time-based** dialog box, select the column name from **Column** to store the timestamp value indicating when each row was added or most recently updated.

In the **Life-time** field, indicate the length of time in which data that has not been updated is to be kept in the table by specifying a numeric value followed by a unit of minutes, hours or days in the drop-down menu.

In the **Cycle** field, indicate the frequency at which data is to be aged out of the table by specifying a numeric value followed by a unit of minutes, hours or days. The default time-based aging cycle is 5 minutes.

Specify an aging state by choosing either **On** or **Off** in the **Time-based aging** field. The default aging state is on. Then click **Apply** to add the time-based aging policy to the table.

Figure 4–25 Adding time-based aging
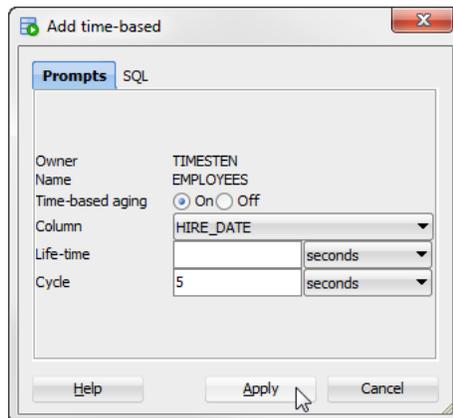
A time-based aging policy can only be added to a table that contains a non-nullable DATE or TIMESTAMP column.

Dropping an aging policy from a table

To drop an existing aging policy from a regular table or a cache table, under the **Tables** node, right-click the name of the table to drop an aging policy from and select **Aging**, then select **Drop**. See Figure 4–23, "Aging attributes".

If you are unable to find the table that you are looking for, see "Locating a TimesTen database object" on page 4-52.

Click **Apply** to drop the aging policy from the table.

Figure 4–26   Drop aging policy dialog

Changing the aging state of a table's aging policy

To change the aging state of a regular table's or cache table's existing aging policy, under the **Tables** node, right-click the name of the table to change the aging state of and select **Aging**, then select **Change State On/Off**. See Figure 4–23, "Aging attributes".

If you are unable to find the table that you are looking for, see "Locating a TimesTen database object" on page 4-52.

In the **Prompts** tab of the **Change state on/off** dialog, change the aging state by selecting either **On** or **Off** in the **Change aging state** field. Click **Apply** to change the aging state of the table.

Figure 4–27   Changing aging state
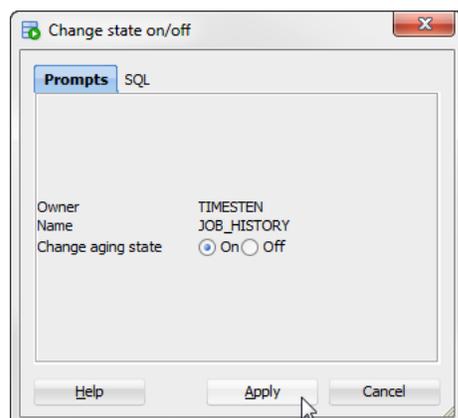
Changing the memory usage thresholds and LRU aging cycle

To change the memory usage thresholds and the LRU aging cycle, right-click the node of the connection name for the TimesTen database and choose **Change Usage-based Aging Attributes**.

Figure 4–28 Change usage-based aging attributes
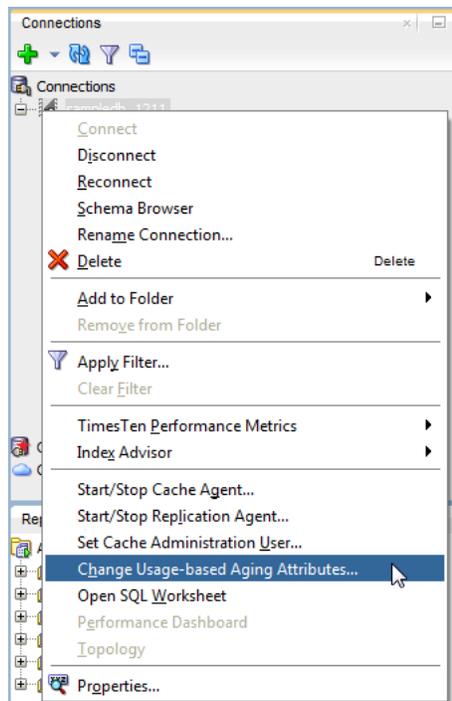
In the **Prompts** tab of the **Changing the usage-based thresholds** dialog box, specify the threshold that determines when data starts being deleted from the tables in the **High usage threshold** field. Specify the threshold that determines when data stops being deleted from the tables in the **Low usage threshold** field. Specify the LRU aging cycle in the **Update Frequency** field. Then click **Apply** to change the memory usage thresholds and the LRU aging cycle.

Figure 4–29 Usage-based aging thresholds dialog
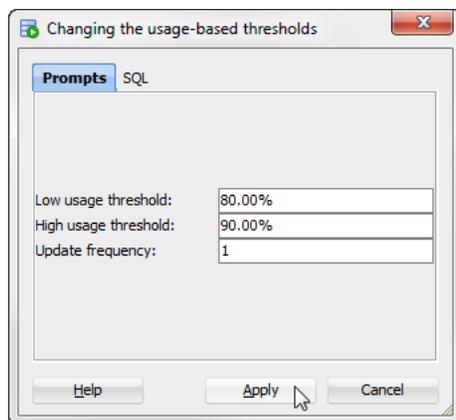
The new settings apply to all tables that have an LRU aging policy defined.

Changing the lifetime and cycle of a table's time-based aging policy

To change the lifetime and cycle of a regular table's or cache table's existing time-based aging policy, under the **Tables** node, right-click the name of the table to change the lifetime and cycle of and select **Aging**, then select **Change Lifetime and Cycle**. See Figure 4–23, "Aging attributes".

If you are unable to find the table that you are looking for, see "Locating a TimesTen database object" on page 4-52 for information.

In the **Life-time** field within the **Prompts** tab of the **Change Lifetime and Cycle** dialog box, change the length of time in which data that has not been updated is to be kept in the table by specifying a numeric value followed by a unit of minutes, hours or days.

In the **Cycle** field, change the frequency at which data is to be aged out of the table by specifying a numeric value followed by a unit of minutes, hours or days.

Then click **Apply** to change the lifetime and cycle for the table.

Figure 4–30   Changing lifetime and cycle
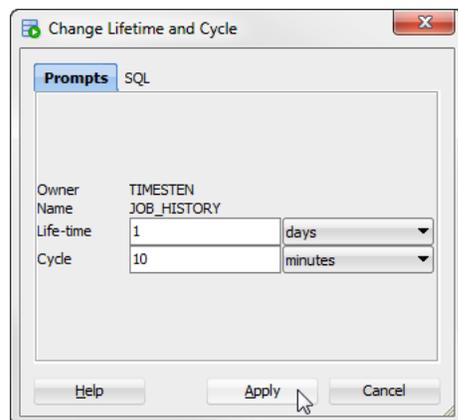
Load data into tables

You can load data into your table using **Import Data**. Before you can load your data, you must export the data into a file. To export the data, right-click on the name of the table and choose **Export**. After successfully exporting your data, you can import the data into a TimesTen table. The TimesTen export feature does not support exporting data from a TimesTen table for the purpose of importing that data into an Oracle database.

1. Choose **+** to the left of the **Tables** node to view the list of tables.

2. Right-click on the table name.

3. Select **Table**, then select **Import Data**.

   An Open dialog appears. Locate the data file to import. This file is the file you previously exported.

4. Choose **Open**.

   The data is loaded into the table.

Compute table size information

If you are using TimesTen Release 11.2.2 or later, you can compute table size information.

To view table size information, you must first compute the table size:

1. Choose **+** to the left of the **Tables** node to view the list of tables.

2. Right-click the name of the table to compute table size information.

3. Select **Table,** then select **Compute Size**.

The **Compute Size** dialog appears. The owner and name of the table are displayed.

Figure 4–31   Compute Size dialog
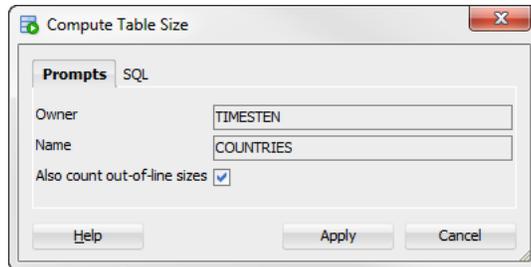
4. Click in the **Also count out-of-line sizes** check box if you want to compute out-of-line sizes.

5. Choose Apply.

A **Confirmation** dialog appears.

6. Choose **OK**.

The table size statistics for the table are computed.

To view the SQL for computing the table size, choose the **SQL** tab in the **Compute Size** dialog. You see that a TimesTen built-in procedure called `ttComputeTabSizes` is executed. After this built-in is executed, you can review the table size information for your table. Note that this table size information is on a per table basis.

To compute table sizes for all tables in your database including materialized views, system tables, and tables that are part of cache groups, use the **SQL Worksheet** and execute the command: `Call ttComputeTabSizes (NULL,0);` or to include out-of-line data: `Call ttComputeTabSizes (NULL,1);`.

After you compute the table size for one or more tables, you can view the table size information:

1. Choose **+** to the left of the **Tables** node to view the list of tables.

2. Choose the table you want to review table size information.

3. Choose the **Sizes** tab located among the tabs that display the attributes of the table.

Table size information is displayed. Specifically, values for `INLINE BYTES`, `OUT OF LINE BYTES`, and `METADATA BYTES` are shown.

Figure 4–32   Displaying table size information

You can choose the **Actions** menu to generate or regenerate the table size. If you select the **Actions** menu, then **Table**, then select **Compute Size**. The **Compute Size** dialog appears allowing you to compute the table size information.

Choose **Refresh** to refresh the displayed table size data.

You can view table size information for all tables that you have computed table sizes. To view such information, select **TimesTen Reports**, then **Table**, then select **Table Sizes**. For more information on TimesTen reports, see "Generating TimesTen Reports" on page 14-1.

For more information on the `ttComputeTabSizes` built-in procedure, see "ttComputeTabSizes" in *Oracle TimesTen In-Memory Database Reference*.

View the characteristics of a table

After you have created a regular table, a global temporary table or a cache table, you can view the characteristics of the table itself such as its columns, indexes and aging policy.

Click the **+** to the left of the **Tables** node to view the list of tables. Click the name of the table to view.

If you are unable to find the table that you are looking for, see "Locating a TimesTen database object" on page 4-52.

Viewing the columns

The **Columns** tab, located within the table's tab, shows information about the columns of a table. The name of the column, data type, nullability property and default value is displayed.

To view the list of columns in a table, locate the **Tables** node. Click the **+** to the left of the node that contains the name of the table.

Figure 4–33   Viewing the columns
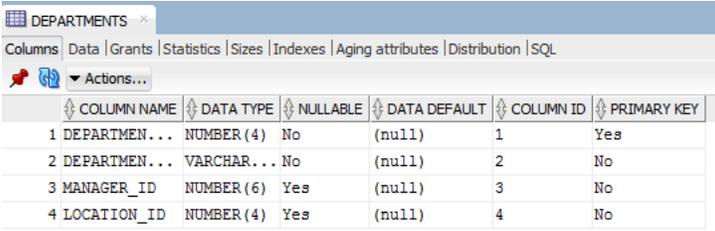
Viewing the indexes

The **Indexes** tab, located within the table tab, shows information about the index for a table. The name of the index, type, uniqueness property, and the number and name of the columns are displayed.

### Viewing the aging attributes

The **Aging attributes** tab, located within the table tab, shows information about the aging attributes for a table. The aging policy type, aging cycle, and aging state are displayed.

For tables that have an LRU aging policy defined, the TimesTen database memory usage thresholds are displayed. For information about how to change these thresholds

and the LRU aging cycle, see "Specifying an aging policy" on page 9-8.

For tables that have a time-based aging policy defined, the name of the non-nullable DATE or TIMESTAMP column used to store the timestamp value (indicating when each row was added or most recently updated) and the length of time non-updated data is not deleted from the table is displayed.

**Figure 4–34   Viewing the aging attributes for a table**
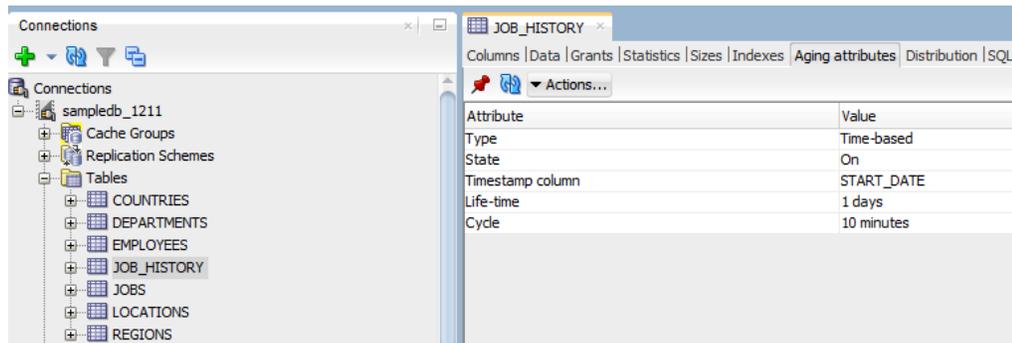
Viewing the table definition

The **SQL** tab, located within the table tab, shows the CREATE TABLE statement that was used to create the table for a regular table or a global temporary table.

Since a cache table is created when its accompanying cache group is created, you can view a cache table's definition by viewing its cache group definition. See "Viewing the cache group definition" on page 11-3.

Figure 4–35   Viewing the SQL for the table definition
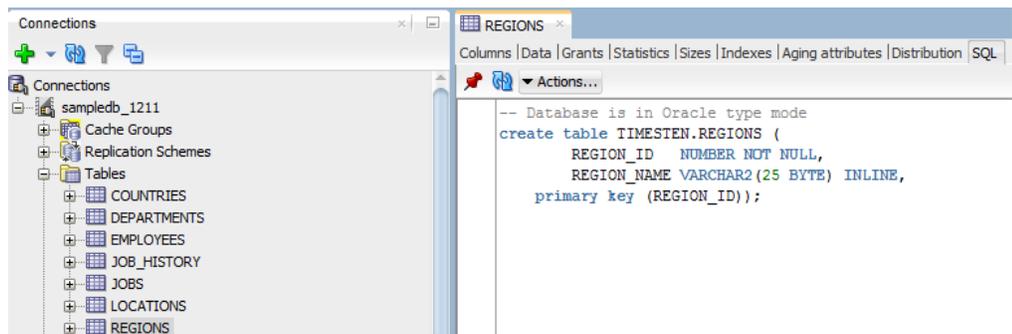
Drop a constraint

You can drop a constraint from a TimesTen table. Ensure that you are on the main SQL Developer page and that your TimesTen connection is expanded.

1.   Click the **+** to the left of the **Tables** node to expand the list of tables.

Figure 4–36 Expand Tables
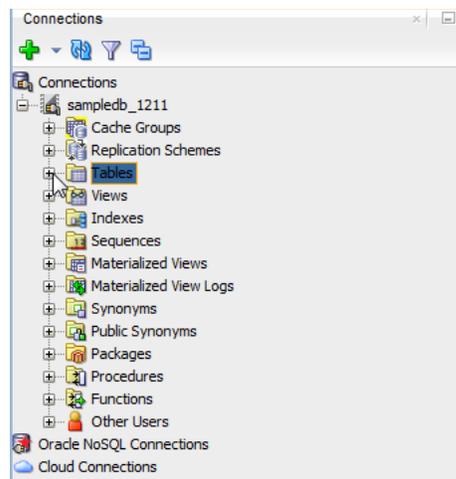
2. Right-click the name of the table and select **Constraint**, then select **Drop**.

Figure 4–37 Select Drop
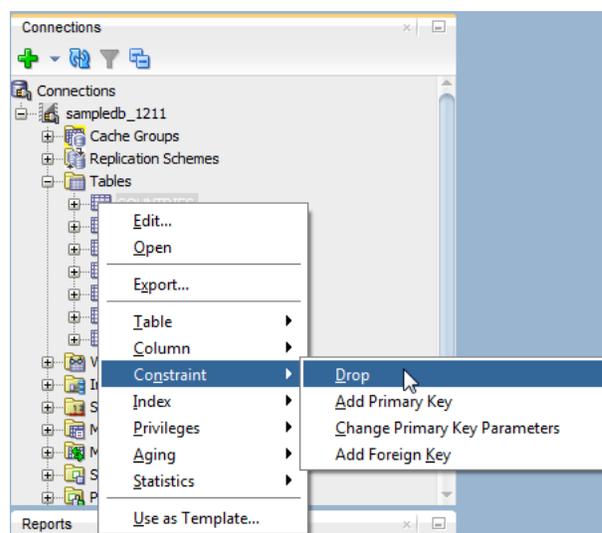
The **Drop** dialog displays. The Owner and Name fields are auto-filled and you cannot edit these fields. Prepare to select the Constraint.

3. Select the constraint that you want to drop from the **Constraint** drop-down list.

To view the SQL statement that will be used to drop the constraint, choose the **SQL** tab of the **Drop** dialog box.

4. Click **Apply**.

Figure 4–38   Drop constraint
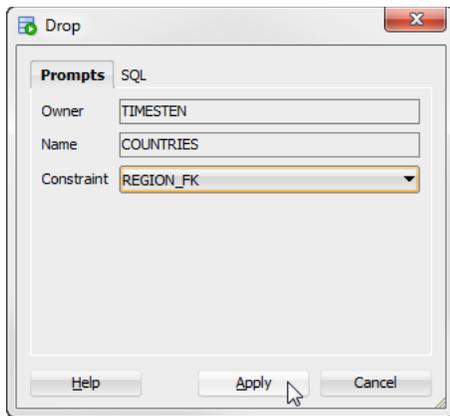
The **Confirmation** dialog box displays indicating that the constraint has been dropped.

5. Click **OK**.

Figure 4–39   Click OK

The constraint has been successfully dropped.

Working with PL/SQL

In SQL Developer you can work with PL/SQL. This section includes these topics:

■ Create a PL/SQL package

■ Define the body of a PL/SQL package

■ Run a PL/SQL package

■ Compile PL/SQL packages

■ Drop a PL/SQL package

■ Use a REF CURSOR as an OUT parameter

Create a PL/SQL package

You can create a PL/SQL package for the TimesTen database with SQL Developer.

To create a PL/SQL package, ensure that you are on the main SQL Developer page and that your TimesTen connection is expanded.

1. Right-click the **Packages** option and select **New Package**.

Figure 4–40   Create a new PL/SQL package
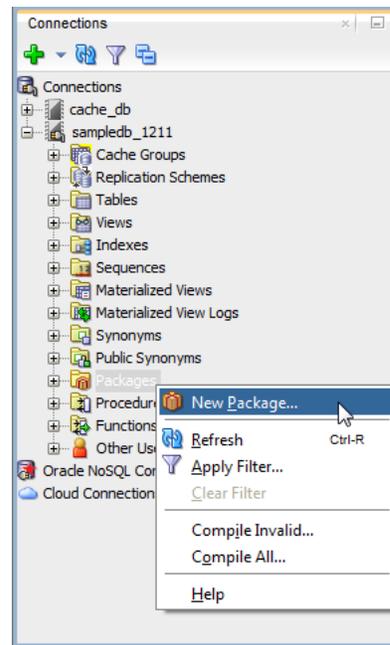
The Create Package dialog displays. Locate the schema drop-down list.

2. Expand the drop-down list labeled **Schema** and select the schema for which you want to create the PL/SQL package.

3. In the **Name** text field, type a name for the PL/SQL package.

4. Optionally, check the **Add New Source In Lowercase** checkbox to create the new source with lowercase text. If you select this option, the text is entered in lowercase regardless of the case in which you type it. This option affects only the appearance of the package name, because PL/SQL is not case sensitive in its execution.

5. Click **OK**.

Figure 4–41   Click OK
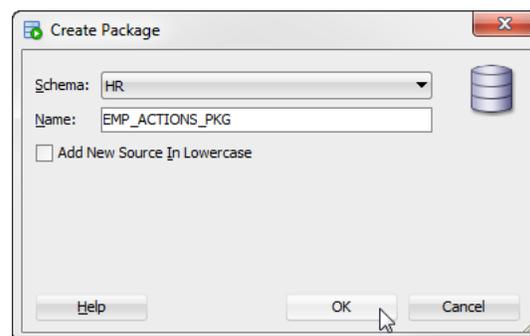
The Create Package dialog closes and the PL/SQL package worksheet displays. Locate the PL/SQL package worksheet.

6. Define the PL/SQL package in the PL/SQL package worksheet.

When you finish defining your PL/SQL package, you can compile and save the PL/SQL package. Locate the File menu in the top-left corner of the SQL Developer window.

7. From the **File** menu, select **Save**.

Figure 4–42 Save the PL/SQL package
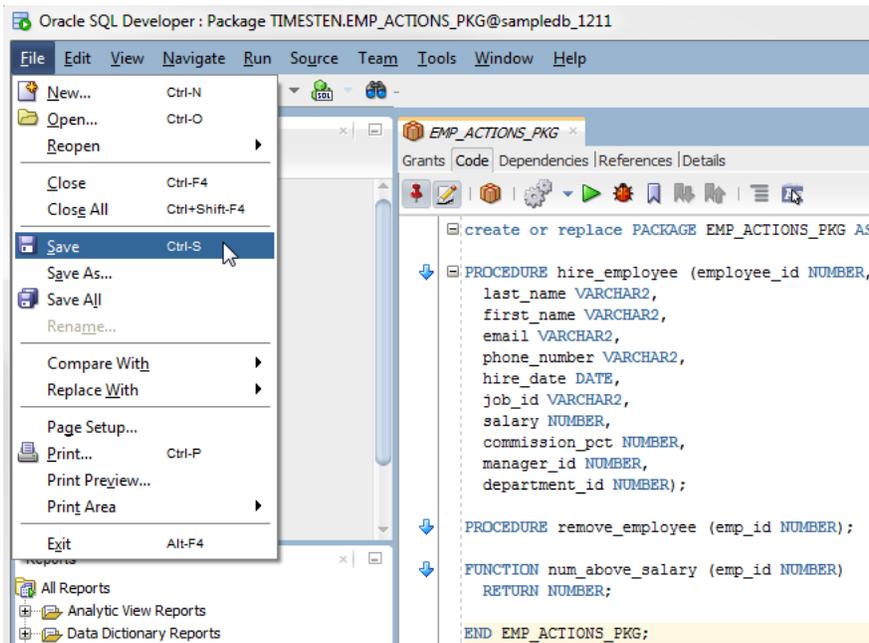
SQL Developer compiles and saves your PL/SQL package. Review the Messages - Log area below the PL/SQL package worksheet to see the compilation results. If you see any compilation errors, troubleshoot and fix the compilation errors. Then, recompile and save your PL/SQL package.

You have successfully compiled and saved a PL/SQL package. You are now ready to specify a PL/SQL package body.

Define the body of a PL/SQL package

Before you define the body of a PL/SQL package, ensure that you have defined a PL/SQL package.

To define the body of a PL/SQL package, ensure that you are on the main SQL Developer page and that your TimesTen connection is expanded.

1. Click **+** to the left of the **Packages** node to view a list of PL/SQL packages.

2. Right-click the **name of your PL/SQL package** and select **Create Body**.

Figure 4–43   Select Create Body
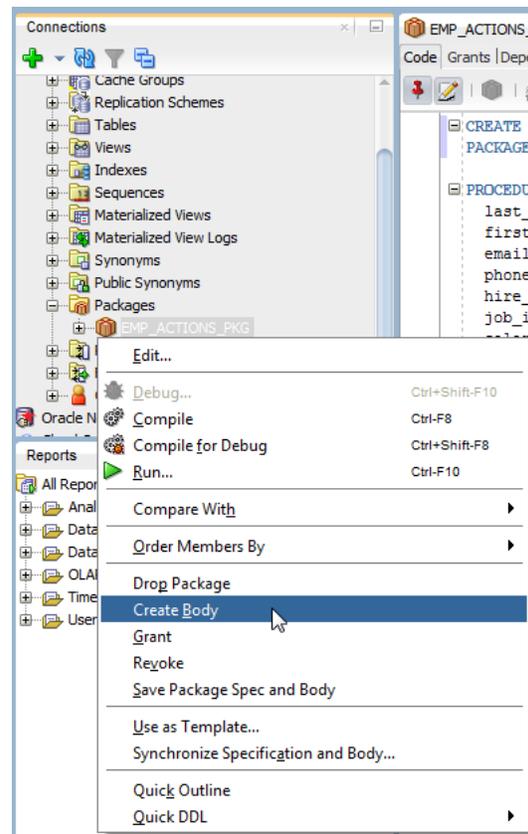
The PL/SQL package body worksheet displays. Locate the PL/SQL package body worksheet.

3.   Define the body of you PL/SQL package in the PL/SQL package body worksheet.

When you finish defining the body of your PL/SQL package, you can compile and save the body of your PL/SQL package. Locate the File menu in the top-left corner of the SQL Developer window.

4.   From the **File** menu, select **Save**.

Figure 4–44   Save the body of the PL/SQL package
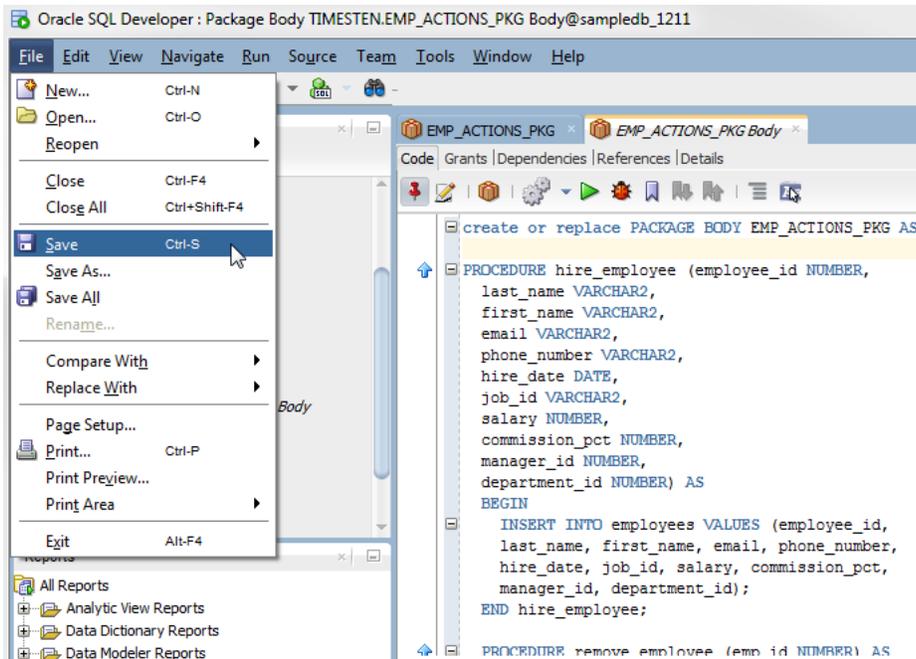
SQL Developer compiles and saves the body of your PL/SQL package. Review the Messages - Log area below the PL/SQL package body worksheet to see the compilation results. If you see any compilation errors, troubleshoot and fix the compilation errors. Then, recompile and save your PL/SQL package.

You have successfully compiled and saved the body of a PL/SQL package. You are now ready to run your PL/SQL package.

Run a PL/SQL package

To run a PL/SQL package, ensure that you are on the main SQL Developer page and that your TimesTen connection is expanded.

1. Click **+** to the left of the **Packages** node to view a list of PL/SQL packages.

2. Right-click the **name of your PL/SQL package** and select **Run**.

Figure 4–45   Select Run
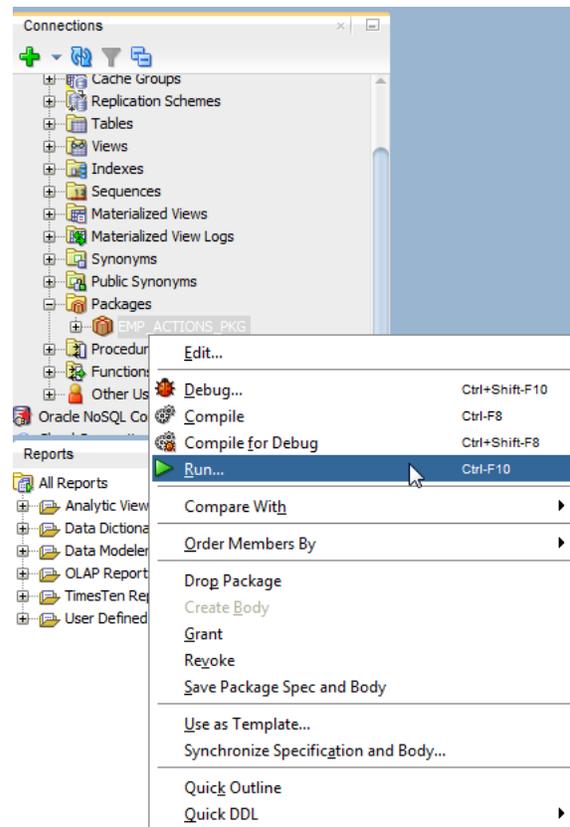
The Run PL/SQL dialog displays. The Run PL/SQL dialog contains three panes:

- Target - This pane shows all the functions and procedures of your PL/SQL package.

- Parameters - This pane shows a list of parameters for the selected function or procedure.

- PL/SQL Block - This pane shows the PL/SQL code that SQL Developer generates to run the selected function or procedure.

In the Target pane, locate the function or procedure that you want to run.

3. In the Target pane, **select the function or procedure** that you want to run.

   The Parameters and PL/SQL Block panes update with information that is relevant to the function or procedure that you selected.

4. In the Parameters pane, **enter values** for all the parameters that are marked as IN mode.

5. In the PL/SQL block, **review the PL/SQL code** and make any necessary changes to run your PL/SQL function or procedure.

6. Click **OK**.

Figure 4–46   Click OK
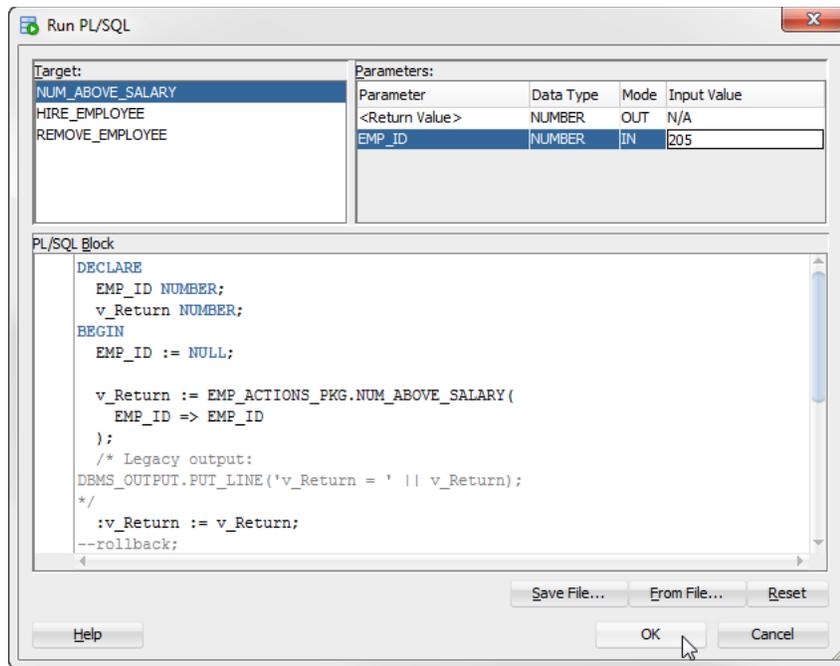
TimesTen runs your PL/SQL procedure or function. Locate the Output Variables tab at the bottom of the main SQL Developer page.

7. Select the **Output Variables** tab.

Figure 4–47   Select the Output Variables tab
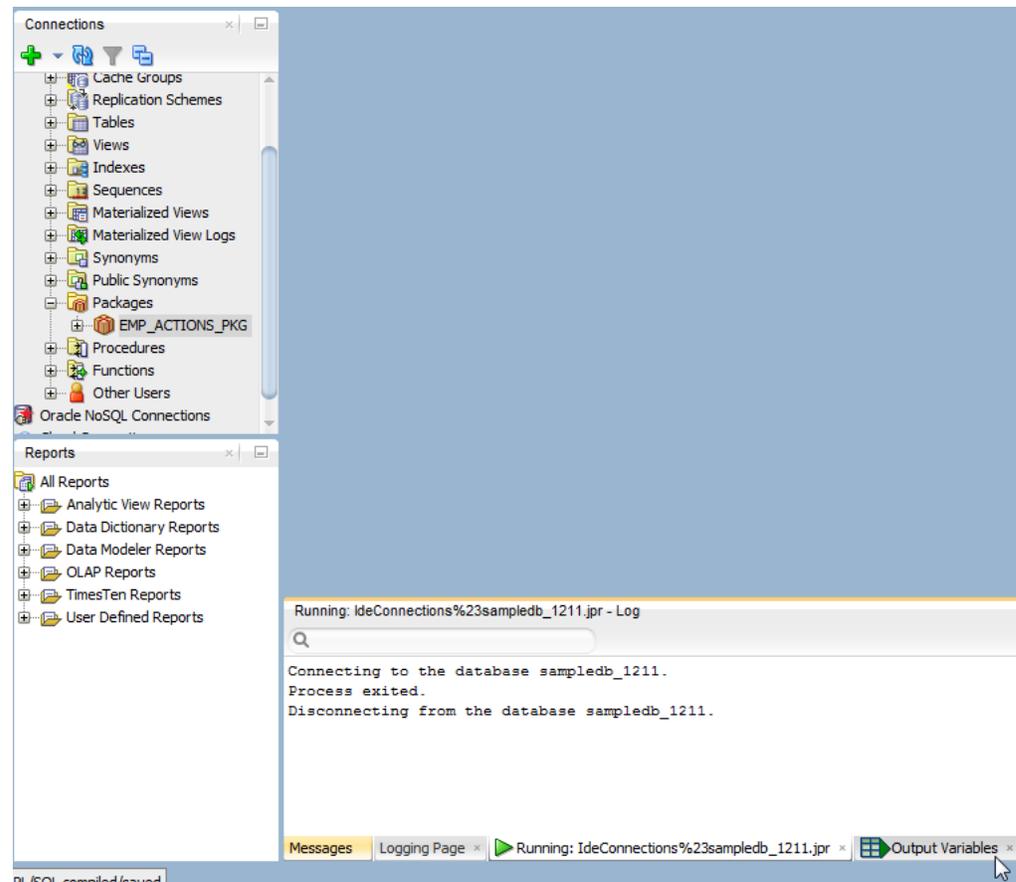
The Output Variables tab displays. Review the Output Variables to see the variable results from the PL/SQL procedure or function.

You successfully ran a PL/SQL package.

Compile PL/SQL packages

Before you compile a PL/SQL package, ensure that you have created a PL/SQL package and defined the body of that PL/SQL package. You should compile a PL/SQL package after you edit any part of your PL/SQL package or package body.

You can either compile a specific PL/SQL package or compile all of your saved PL/SQL packages.

■   Compile a specific PL/SQL package

■   Compile all of your PL/SQL packages

Compile a specific PL/SQL package

To compile a specific PL/SQL package, ensure that you are on the main SQL Developer page and that your TimesTen connection is expanded.

1.   Click **+** to the left of the **Packages** node to view a list of PL/SQL packages.

2.   Right-click the **name of your PL/SQL package** and select **Compile**.

**Figure 4–48   Compile a PL/SQL package**
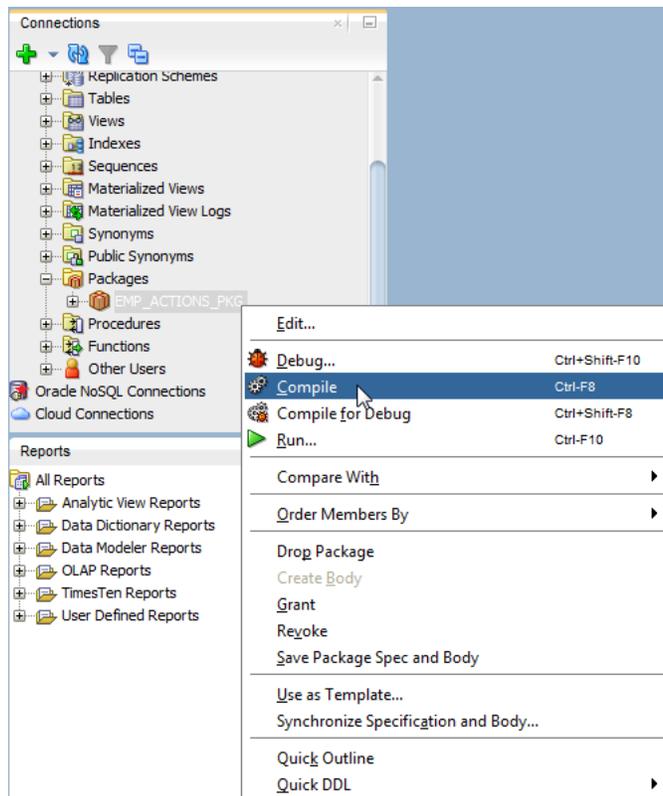
SQL Developer compiles and saves the body of your PL/SQL package. Review the Messages - Log area to see the compilation results. If you see any compilation errors, troubleshoot and fix the compilation errors. Then, recompile and save your PL/SQL package.

You have successfully compiled a PL/SQL package.

Compile all of your PL/SQL packages

To compile all of your saved PL/SQL packages, ensure that you are on the main SQL Developer page and that your TimesTen connection is expanded.

1.   Right-click the **Packages** option and select **Compile All**.

Figure 4–49   Compile all PL/SQL packages
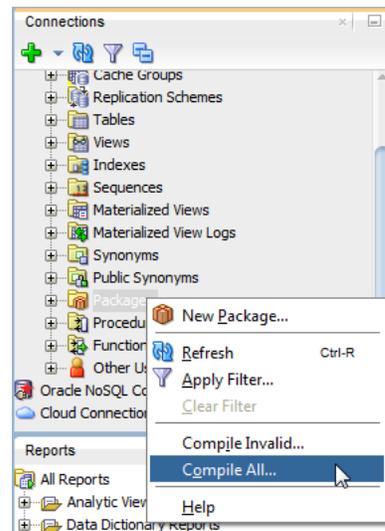
The Compile All dialog displays. Locate the Apply button.

2. Click **Apply**.

TimesTen compiles all of your saved PL/SQL packages. A confirmation dialog displays with the compilation results and any errors that TimesTen encountered.

3. Click **OK**.

Review the Messages - Log area to see the compilation results. If you see any compilation errors, troubleshoot and fix the compilation errors. Then, recompile and save your PL/SQL packages.

You have successfully compiled all of your saved PL/SQL packages.

Compile invalid PL/SQL packages

To compile all of your invalid PL/SQL packages, ensure that you are on the main SQL Developer page and that your TimesTen connection is expanded.

1. Right-click the **Packages** option and select **Compile Invalid**.

Figure 4–50   Select Compile Invalid
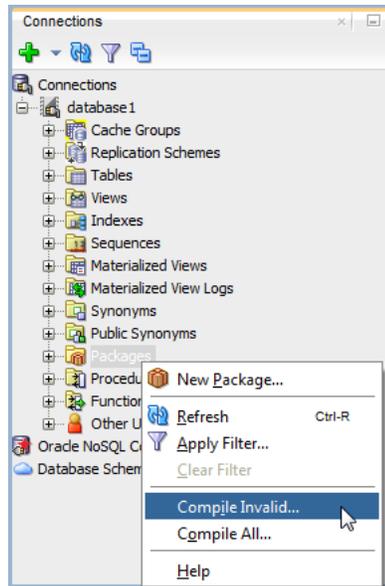
The Compile Invalid dialog displays. Locate the Apply button.

2. Click **Apply**.

TimesTen compiles all of your invalid PL/SQL packages. A confirmation dialog displays with the compilation results and any errors that TimesTen encountered.

3. Click **OK**.

Review the Messages - Log area to see the compilation results. If you see any compilation errors, troubleshoot and fix the compilation errors. Then, recompile and save your PL/SQL packages.

You have successfully compiled all of your invalid PL/SQL packages.

Drop a PL/SQL package

To drop a PL/SQL package, ensure that you are on the main SQL Developer page and that your TimesTen connection is expanded.

1. Click **+** to the left of the **Packages** node to view a list of PL/SQL packages.

2. Right-click the **name of your PL/SQL package** and select **Drop Package**.

Figure 4–51 Drop a PL/SQL package
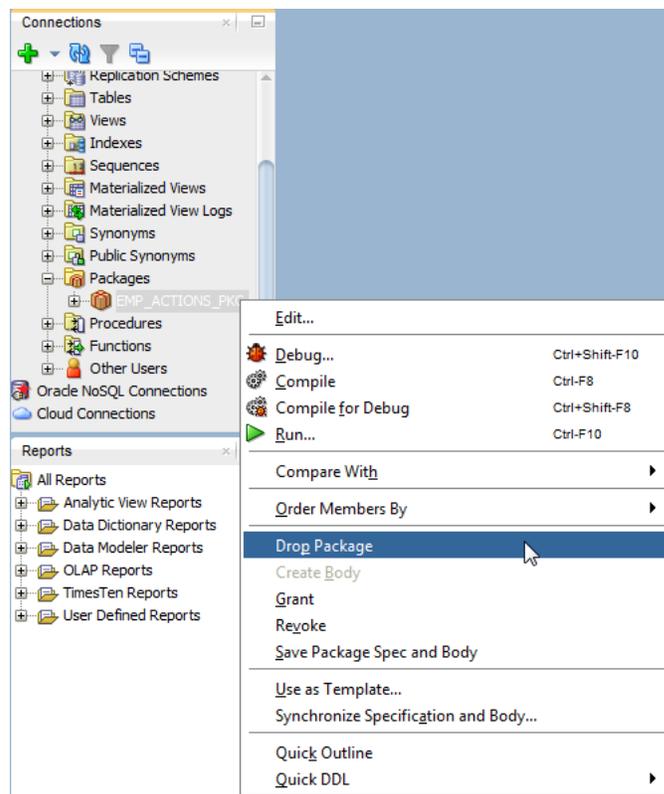
The Drop Package displays. Review that the Owner and Name fields match the information of the PL/SQL package that you want to drop. Locate the Apply button.

3. Click **Apply**.

Figure 4–52 Click Apply
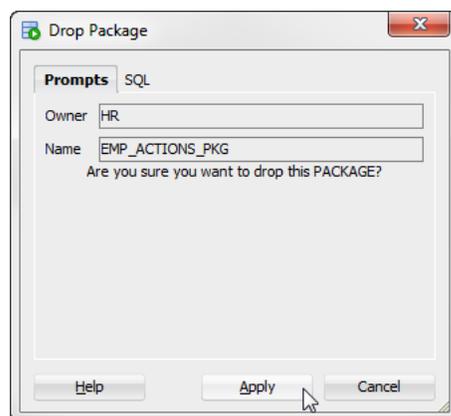
TimesTen drops your PL/SQL package and SQL Developer displays a confirmation dialog.

4. Click **OK**.

The confirmation dialog closes.

You have successfully dropped a PL/SQL package.

Use a REF CURSOR as an OUT parameter

You can test a PL/SQL function, procedure, or package by defining a REF CURSOR as an OUT parameter in your PL/SQL function, procedure, or package. After you define a REF CURSOR, compile and run your PL/SQL function, procedure, or package. The **Run** dialog appears and when you choose **OK**, the details of the run are displayed and the output from the execution of the function, procedure, or package is displayed in the **Output Variables** tab.

The following example creates a package called get_emp_pkg. The get_emp_pkg package defines a REF CURSOR as an OUT parameter and defines a procedure that uses the REF CURSOR as an OUT parameter.

Figure 4–53   get_emp_pkg definition
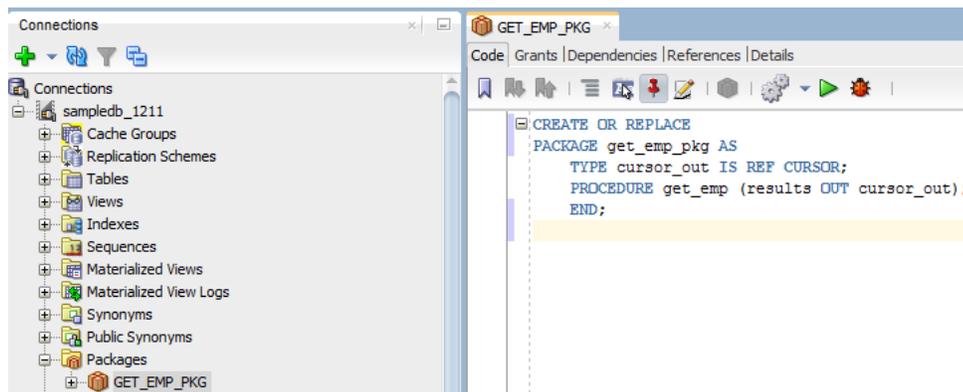
Next create a package body that defines the procedure get_emp. The procedure get_emp opens the REF CURSOR variable and performs a query on the employees table.

Figure 4–54   get_emp_pkg package body definition
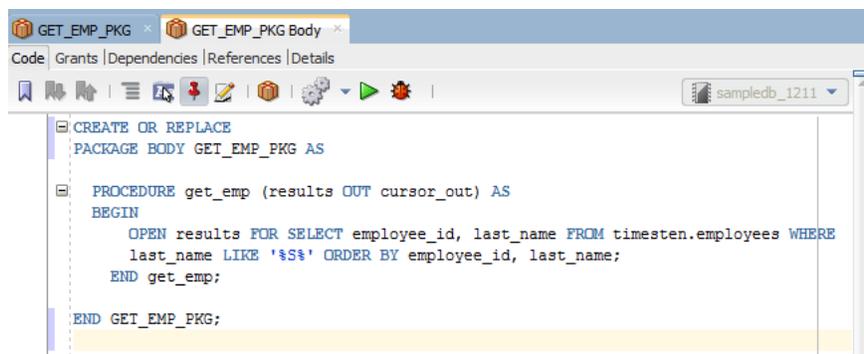
Compile the package and package body. Right-click on the get_emp_pkg and choose **Run**. The **Run PL/SQL** dialog displays allowing you to run the test wrapper. Choose **OK** to run the test wrapper.

Figure 4–55   Viewing the Run PL/SQL dialog
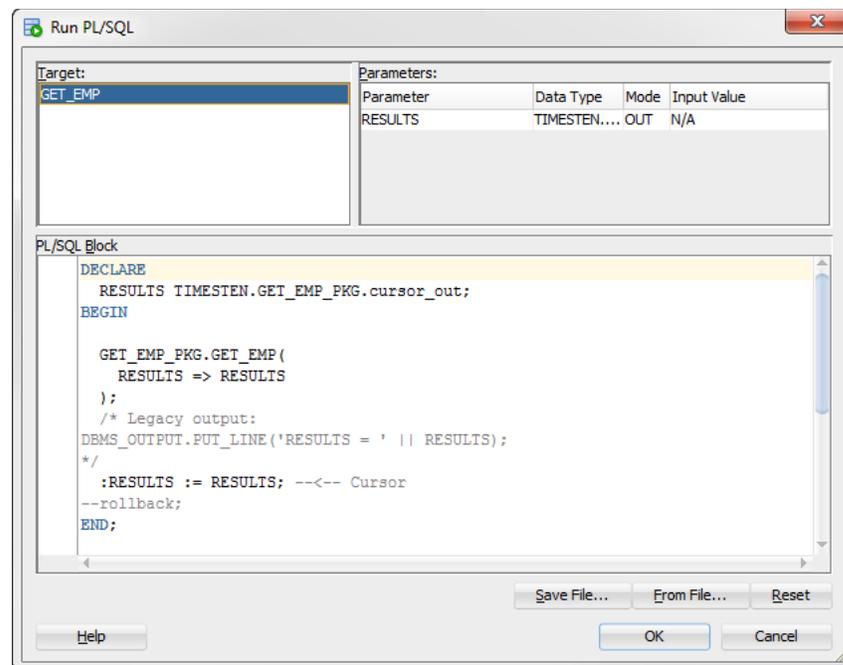
At the bottom of SQL Developer, you see the following tabs:

- **Messages**: Displays the status of all of your compilations and if any errors occurred.

- **Logging Page**: Displays all errors that have occurred.

- **IdeConnections**: Displays the status of your run including any errors encountered.

- **Output Variables**: Shows the output from the execution of the function, procedure, or package.

Figure 4–56 shows the output from the execution of the get_emp_pkg package.

Figure 4–56 Viewing the Output Variables tab
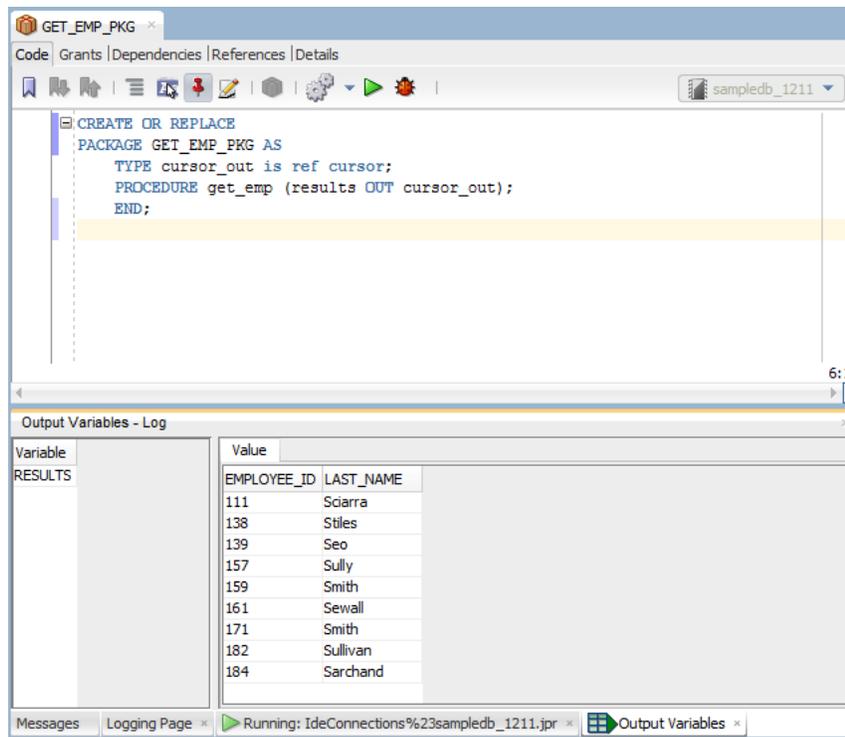
Working with sequences

SQL Developer enables you to work with sequences for TimesTen and TimesTen Scaleout databases. For more information on sequences, see "CREATE SEQUENCE" in the *Oracle TimesTen In-Memory Database SQL Reference*.

This section includes these topics:

- Create a sequence
- Alter a sequence
- Drop a sequence

Create a sequence

You can create a sequence for the TimesTen database or TimesTen Scaleout database with SQL Developer.

To create a sequence, ensure that you are on the main SQL Developer page and that your TimesTen or TimesTen Scaleout connection is expanded.

1. Right-click the **Sequences** option and select **New Sequence**.

Figure 4–57 Select New Sequence
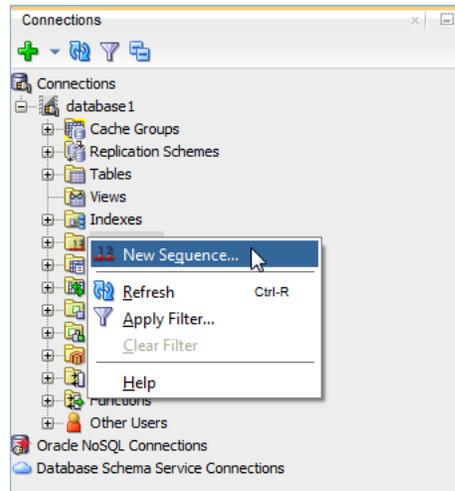
The Create Sequence dialog displays. Locate the Schema drop-down list.

2. From the **Schema** drop-down list, select the owner of the sequence.

3. In the **Name** text field, type a name for the sequence.

4. In the **Start With** text field, specify the first sequence number to be generated.

Use this field to start an ascending sequence at a value that is greater than the minimum value or to start a descending sequence at a value less than the maximum.

5. In the **Increment** text field, specify the incremental value between consecutive numbers.

This value can be either a positive or negative integer. It cannot be 0. If the value is positive, it is an ascending sequence. If the value is negative, it is descending. The default value is 1.

6. In the **Min Value** text field, specify the minimum value for an ascending sequence, or the final value for a descending sequence.

This value must be less than or equal to the value that you specified in the Start With field. The default minimum value is 1.

7. In the **Max Value** text field, specify the largest possible value for an ascending sequence, or the starting value for a descending sequence.

This value must be greater than or equal to the value that you specified in the Start With field. The default maximum value is (2^{63}) -1, which is the maximum value of the `BIGINT` data type.

8. If you want to cache the numbers from this sequence, select the `Cache` option from the **Cache** drop-down list. If not, select the `No Cache` option.

9. If you selected the `Cache` option from the Cache drop-down list, in the **Cache Value** text field, specify the range of numbers that are cached.

When a restart occurs, unused cached numbers are lost. If you specify a cache value of 1, then each use of the sequence results in an update to the database. Larger cache values result in fewer changes to the database and less overhead. The default value is 20.

10. From the **Cycle** drop-down list, select one of these options:

- `Cycle` - The sequence number generator continues to generate numbers after it reaches the maximum or minimum value.

- `No Cycle` - The sequence number generator stops generating numbers when the maximum or minimum value is reached.

By default, sequences do not cycle. Once the number reaches the maximum value in the ascending sequence, the sequence wraps around and generates numbers from its minimum value. For a descending sequence, when the minimum value is reached, the sequence number wraps around, beginning from the maximum value.

11. In the **Batch** value text field, define the range of unique sequence values that are stored at each element of the grid. The default value is 10 million. This option is only available for TimesTen Scaleout databases.

Locate the OK button.

Note: If you want to review the DDL statement that TimesTen or TimesTen Scaleout uses to create your sequence, click the **DDL** tab.

---

12. Click **OK**.

Figure 4–58   Click OK
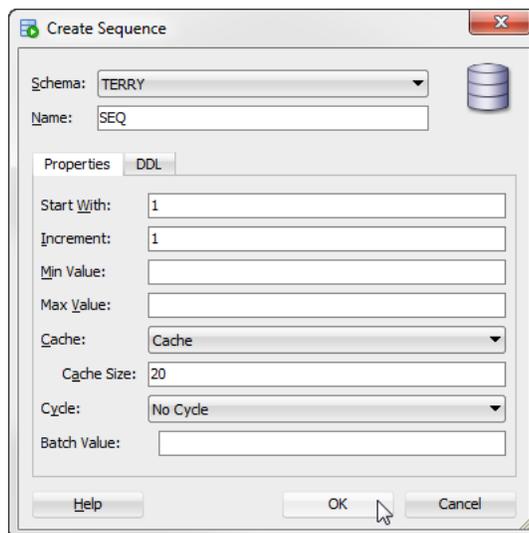
The Create Sequence dialog closes and TimesTen or TimesTen Scaleout creates your sequence.

You have successfully created a sequence.

Alter a sequence

You can alter the batch value of an existing sequence of a TimesTen Scaleout database with SQL Developer.

To alter a sequence, ensure that you are on the main SQL Developer page and that your TimesTen or TimesTen Scaleout connection is expanded.

1. Expand the **Sequences** option, right-click the **name of the sequence**, then select **Edit**.

Figure 4–59   Select Edit
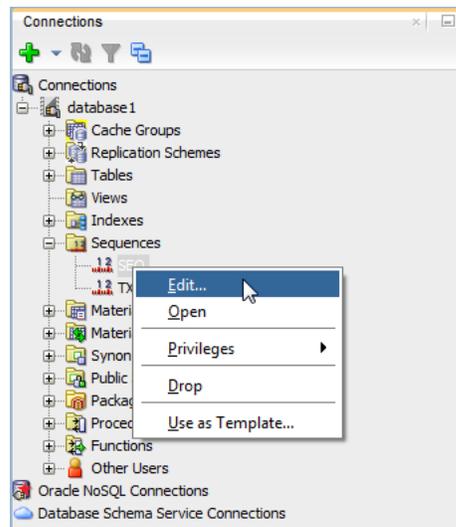
The Edit Sequence dialog displays. Locate the Batch Value text field.

2. In the **Batch Value** text field, type the new batch value for your sequence.

Locate the OK button.

3. Click **OK**.

The Edit Sequence dialog closes.

You have successfully edited the batch value of an existing sequence.

Drop a sequence

You can drop a sequence of the TimesTen database or TimesTen Scaleout database with SQL Developer.

To drop a sequence, ensure that you are on the main SQL Developer page and that your TimesTen or TimesTen Scaleout connection is expanded.

Expand the **Sequences** option, right-click the **name of the sequence**, then select **Drop**.

Figure 4–60   Select Drop
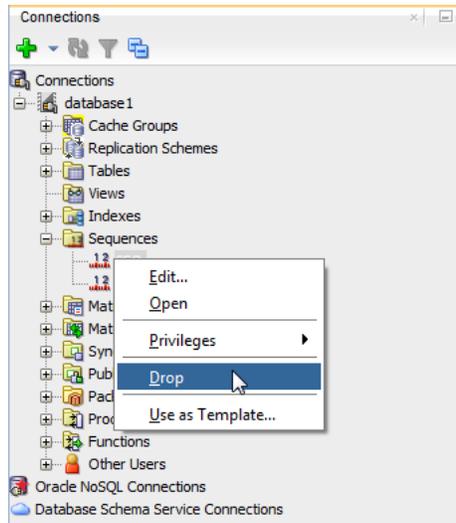
The Drop dialog displays. Locate the Apply button.

Click **Apply**.

The Drop dialog closes and a Confirmation dialog closes. Locate the OK button.

Click **OK**.

The Confirmation dialog closes.

You have successfully dropped a sequence from your database.

Working with views

SQL Developer enables you to work with views for the TimesTen and TimesTen Scaleout databases. For more information on views, see "Understanding views" in the *Oracle TimesTen In-Memory Database Operations Guide*.

This section includes these topics:

- Create a view
- Drop a view

Create a view

You can create a view for the TimesTen database or TimesTen Scaleout database with SQL Developer.

To create a view, ensure that you are on the main SQL Developer page and that your TimesTen or TimesTen Scaleout connection is expanded.

1. Right-click the **Views** option and select **New View**.

Figure 4–61   Select New View
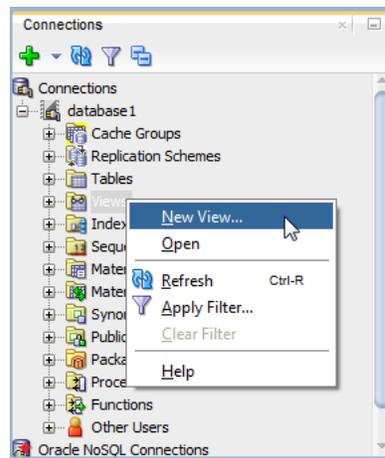
The New View dialog displays. Locate the View Name text field.

2.   In the **View Name** text field, type a name for the view.

3.   In the **SQL** text field, type the SQL query for the view.

4.   Click **Apply**.

Figure 4–62   Click Apply
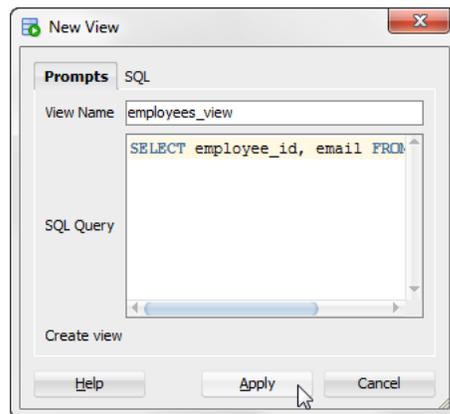
The New View dialog closes and a Confirmation dialog displays.

5.   Click **OK**.

The Confirmation dialog closes and TimesTen creates your view.

You have successfully created a view.

Drop a view

You can drop a view of the TimesTen database or TimesTen Scaleout database with SQL Developer.

To drop a view, ensure that you are on the main SQL Developer page and that your TimesTen or TimesTen Scaleout connection is expanded.

Expand the **Views** option, right-click the **name of the view**, then select **Drop**.

Figure 4–63   Select Drop
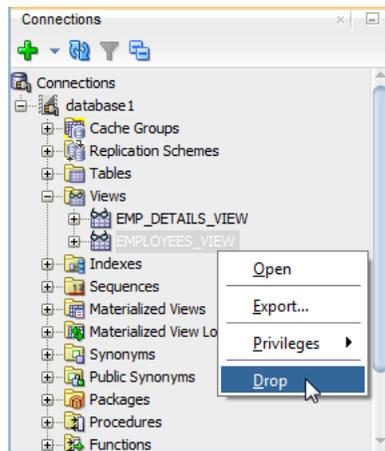
The Drop dialog displays. Locate the Apply button.

Click **Apply**.

The Drop dialog closes and a Confirmation dialog closes. Locate the OK button.

Click **OK**.

The Confirmation dialog closes.

You have successfully dropped a view from your database.

Working with materialized views

SQL Developer enables you to work with materialized views for the TimesTen and TimesTen Scaleout databases. For more information on views, see "Understanding materialized views" in the *Oracle TimesTen In-Memory Database Operations Guide*.

This section includes these topics:

- Create a materialized view
- Drop a materialized view

Create a materialized view

You can create a materialized view for the TimesTen database or TimesTen Scaleout database with SQL Developer.

To create a materialized view, ensure that you are on the main SQL Developer page and that your TimesTen or TimesTen Scaleout connection is expanded.

1. Right-click the **Materialized Views** option and select **New Materialized View**.

Figure 4–64   Select New Materialized View
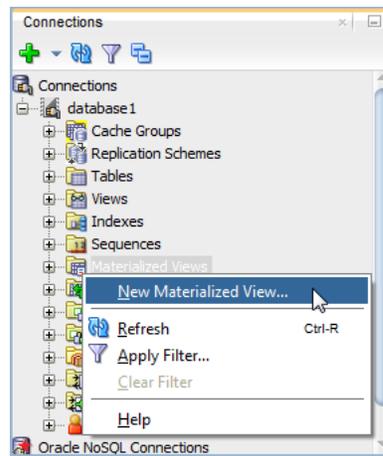
The New Materialized View dialog displays. Locate the Materialized View Name text field.

2.   In the **Materialized View Name** text field, type a name for the view.

3.   In the **SQL** text field, type the SQL query for the view.

4.   Click **Apply**.

Figure 4–65   Click Apply
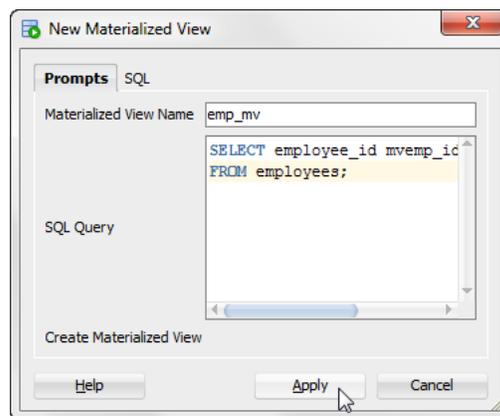
The New Materialized View dialog closes and a confirmation dialog displays.

5.   Click **OK**.

The Confirmation dialog closes and TimesTen creates your view.

You have successfully created a materialized view.

Drop a materialized view

You can drop a materialized view of the TimesTen database or TimesTen Scaleout database with SQL Developer.

To drop a view, ensure that you are on the main SQL Developer page and that your TimesTen or TimesTen Scaleout connection is expanded.

Expand the **Materialized Views** option, right-click the **name of the materialized view**, then select **Drop**.

Figure 4–66   Select Drop
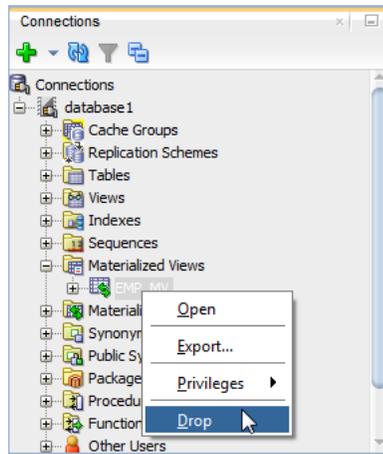
The Drop dialog displays. Locate the Apply button.

Click **Apply**.

The Drop dialog closes and a Confirmation dialog closes. Locate the OK button.

Click **OK**.

The Confirmation dialog closes.

You have successfully dropped a materialized view from your database.

Locating a TimesTen database object

If you are unable to find a database object, right-click the node corresponding to the object type and select **Refresh** to refresh the list of objects of that type.

For example, to find a cache group, right-click the **Cache Groups** node and select **Refresh** to refresh the list of cache groups.

You can also check if the object appears under the **Other Users** node. Click the **+** to the left of the **Other Users** node to expand the node. Then click the **+** to the left of any of the users' nodes to expand that user's node.

Click the **+** to the left of the node corresponding to the object type to view the list of objects of that type owned by the user.

For example, to find a table, click the **+** to the left of the user's **Tables** node to view the list of tables owned by that user.

Granting and revoking object privileges

You can grant privileges to and revoke privileges from the following database objects:

- Regular and cache tables

- Regular and materialized views

- Sequences

- PL/SQL packages, procedures and functions

To grant privileges on an object, right-click the name of the object and select
Privileges, then select **Grant**.

Figure 4–67   Granting and revoking privileges
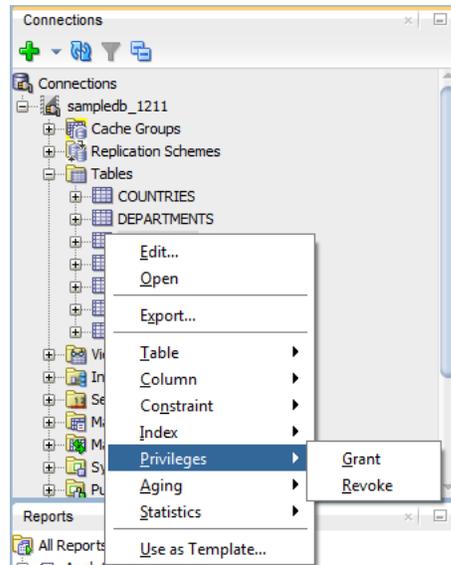
In the **Users** drop-down menu of the **Grant** dialog box, select the user to grant object
privileges to the object. In the **Privileges** section, select the **All** check box or click **>>** to
grant all available object privileges on the object to the selected user. Otherwise, select
the individual privileges from the **Available Privileges** list for the privileges to grant
to the selected user and then click **>** to move those privileges into the **Selected
Privileges** list. To select multiple privileges, press and hold the CTRL key, and click the
desired privileges. Click **Apply** to grant the selected object privileges on the object to
the selected user.

Figure 4–68   Grant privileges dialog
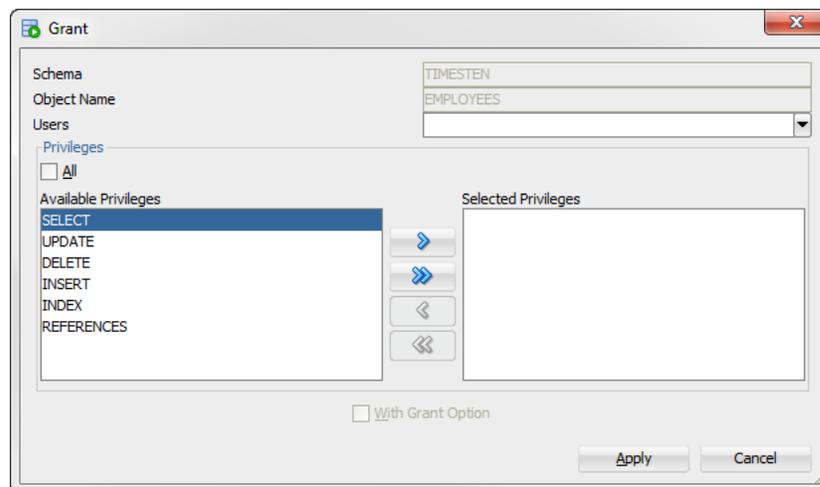
For a particular object type, only the available object privileges are shown. For
example, `DELETE`, `INDEX`, `INSERT`, `REFERENCES`, `SELECT` and `UPDATE` privileges can
be granted on a table. However, only `INDEX`, `REFERENCES` and `SELECT` privileges can
be granted on a materialized view.

Only users with the ADMIN system privilege or the owner of an object can grant object privileges on the object.

For more information about the set of privileges available to each type of object, see "Object privileges" in the *Oracle TimesTen In-Memory Database SQL Reference*.

To revoke privileges from an object, right-click the name of the object and select **Privileges**, then select **Revoke**. See Figure 4–67, "Granting and revoking privileges".

In the **Users** drop-down menu of the **Revoke** dialog box, select the user to revoke object privileges from the object. In the **Privileges** section, select the **All** check box or click **>>** to revoke all granted object privileges on the object from the selected user. Otherwise, select the individual privileges from the **Available Privileges** list for the privileges to revoke from the selected user and then click **>** to move those privileges into the **Selected Privileges** list. To select multiple privileges, press and hold the CTRL key, and click the desired privileges. Click **Apply** to revoke the selected object privileges on the object from the selected user.

Figure 4–69   Revoke privileges dialog
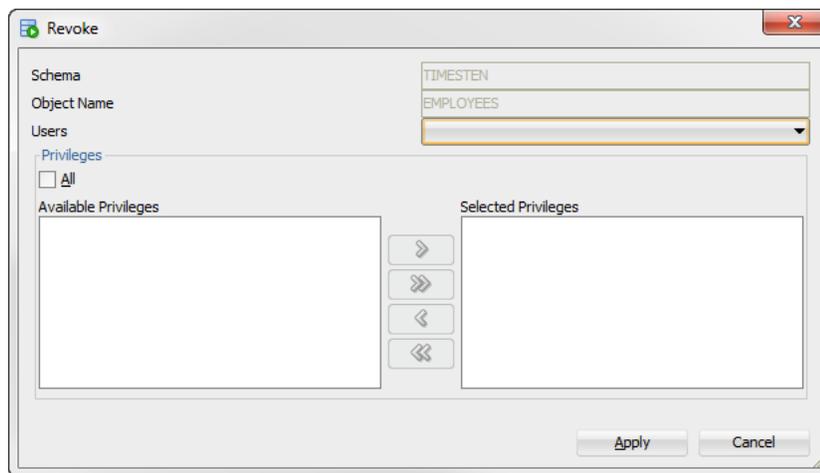
The **Users** drop-down menu shows only users that have object privileges on the object. For a particular user, only the object privileges that they have been granted are shown.

Only users with the ADMIN system privilege or the owner of an object can revoke object privileges from the object.

Viewing the privileges granted on a database object

You can view the object privileges granted to all users on a particular object such as a regular table or a cache table, a regular view or a materialized view, a sequence, or a PL/SQL package, procedure or function.

For example, to view the privileges granted on a regular table, click the **+** to the left of the user's **Tables** node to view the list of tables owned by that user and then click the name of the desired table.

The **Grants** tab, located within the table tab, shows the users who have privileges on the table, what privileges they have been granted, and the user who granted the privileges.

ttIsql commands in SQL worksheet

The following *ttIsql* commands are supported in the SQL Worksheet:

- autocommit

- desc

- dssize

- version

For more information, see "ttIsql" in the *Oracle TimesTen In-Memory Database Reference*.

5

Working with the TimesTen Query Optimizer

This chapter shows you how to interact with the TimesTen query optimizer in SQL Developer. Topics include:

- Updating the table and column statistics used by the query optimizer
- Exporting table statistics
- Viewing the execution plan of a statement

Updating the table and column statistics used by the query optimizer

The query optimizer uses statistics stored in TimesTen database system tables to determine the optimal execution plan for a statement. TimesTen stores table level statistics in the SYS.TBL_STATS table. Column level statistics, such as the minimum and maximum value, and the number of unique values in a column, are stored in the SYS.COL_STATS table.

TimesTen does not compute table and column statistics as updates occur on regular and cache tables. Instead, TimesTen updates statistics when an explicit request is made.

You can update statistics for all tables owned by a user or for a specific table owned by an user.

To update statistics for all tables, right-click the **Tables** node and select **Statistics**, then select **Update**.

To update statistics for a specific table, click the **+** to the left of the **Tables** node. Right-click the name of the table and select **Statistics**, then select **Update**.

Figure 5–1   Update statistics for a table
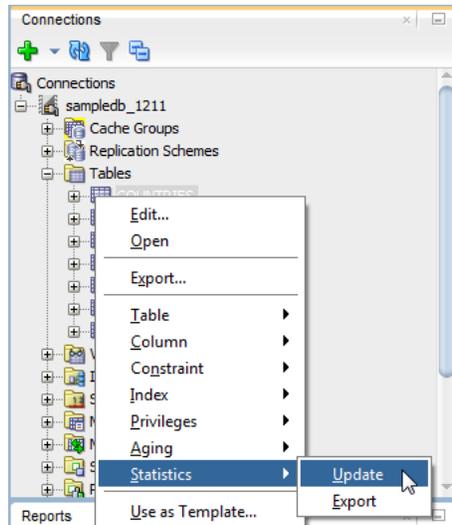
If you are unable to find the table, see "View the characteristics of a table" on page 4-27.

In the **Prompts** tab of the **Update statistics** dialog, choose **Invalidate referenced commands** to invalidate the execution plans of statements that reference the table for which statistics are updated. When you invalidate the execution plan of a statement, TimesTen recompiles or reprepares that statement upon its next execution.

For **Interval type for table statistics**, choose **Complete interval** to divide the rows of the table into two or more intervals and compute statistics on each interval, or **Single interval** to compute statistics on the entire set of rows as a single interval. You must define a range index on the table to compute complete interval statistics.

Click **Apply**.

Figure 5–2   Update statistics dialog
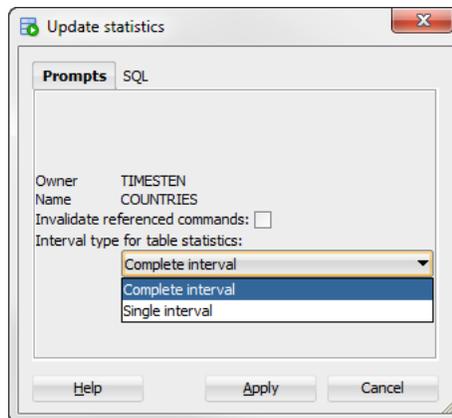
To view the optimizer statistics of a table, click the name of the table.

Figure 5–3   Statistics for employees table

If you are unable to find the table that you are looking for, see "Locating a TimesTen database object" on page 4-52.

In the **Statistics** tab of the table pane, the top section shows the table statistics such as:

- The number of rows in the table

- A timestamp indicating when statistics were most recently updated for the table

The bottom section shows the column statistics. For each column, the following information is displayed:

- The name of the column

- The number of intervals the data is divided into to compute statistics

- Total number of NULL values

- Total number of non-NULL unique values

- Total number of rows in each interval

For each interval of each column, SQL Developer displays the following:

- The number of unique values other than the most frequently occurring value

- The number of rows that contain a value other than the most frequently occurring value

- The number of rows that contain the most frequently occurring value

- The minimum value

- The maximum value

- The most frequently occurring value

For more information about query optimizer statistics, see "Statistics" in the *Oracle TimesTen In-Memory Database Introduction*, "When optimization occurs" in the *Oracle TimesTen In-Memory Database Operations Guide* or "ttOptUpdateStats" in the *Oracle TimesTen In-Memory Database Reference*.

Exporting table statistics

You can generate a set of SQL statements that allow you to restore the table statistics to the current state. You can generate SQL statements for a single table or for user tables that the connection user has permission to access. After you export your table statistics to a `.sql` file, you can import your table statistics by opening the `.sql` file in SQL Developer and running the script file.

To export the current TimesTen table statistics, ensure that you are on the main SQL Developer page.

1. To save table statistics for a single table, right-click the name of the table and select **Statistics**, then select **Export**. To save table statistics for all user tables, right-click **Tables** and select **Statistics**, then select **Export**.

Figure 5–4   Exporting statistics
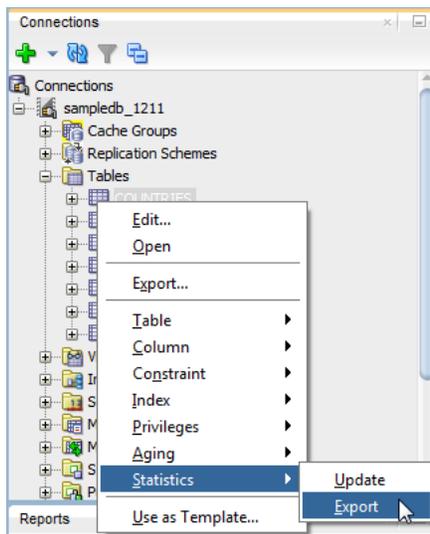
The **Export Optimizer Statistics** dialog displays. Locate the **Save** button at the bottom right of the dialog.

2. Click **Save**.

Figure 5–5   Click save
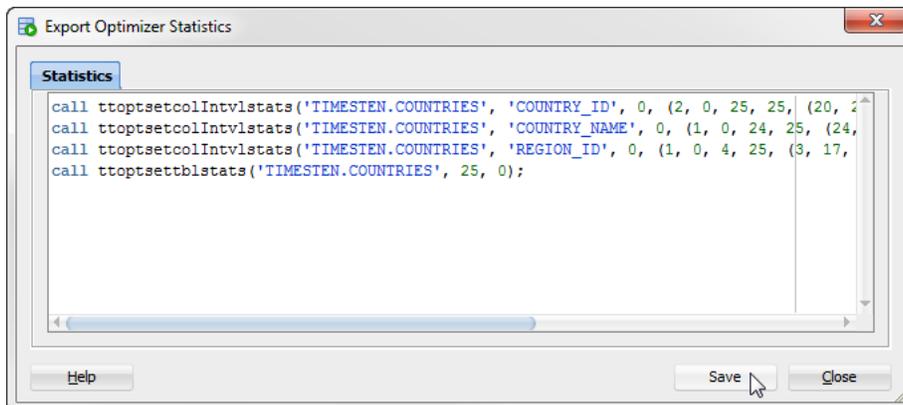
The **Save Optimizer Statistics Export Commands** dialog displays.

3. Select the directory where you want to save the current TimesTen table statistics.

4. In the **File Name** field, define the name of your current TimesTen table statistics.

   The default filename of the current TimesTen table statistics is `connection_name-exportstatistics-YYYYMMDDMISS.sql`, where `connection_name` is the name of the connection. `YYYYMMDDHHMISS` is a timestamp of when the recommendations were made, where `YYYY` is the year, `MM` the month, `DD` the day, `HH` the hour, `MI` the minutes, and `SS` the seconds.

5. Click **Save**.

Figure 5–6   Save current TimesTen table statistics
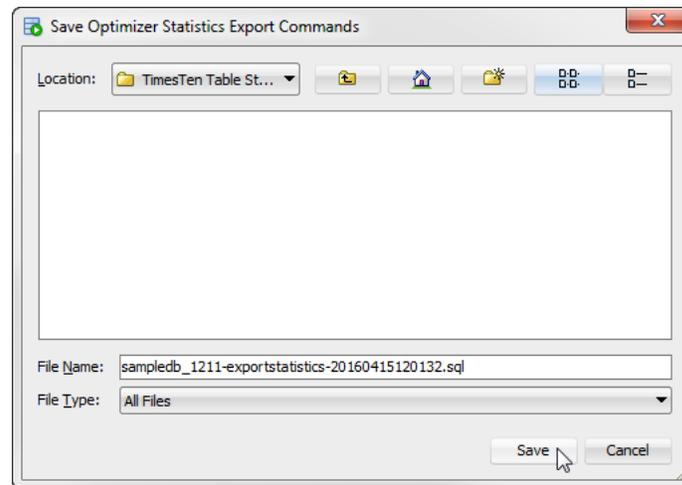
TimesTen saves your current TimesTen table statistics.

# Viewing the execution plan of a statement

A statement must be prepared or compiled before it is executed. When the statement is prepared or compiled, the query optimizer generates an execution plan for the statement.

To view the execution plan of a statement, create the statement in the SQL Worksheet. If you do not have a worksheet open in your connection, right-click the node of the connection name and select **Open SQL Worksheet**. See Figure 3–7, "Viewing a TimesTen connection".

If you do not have a connection open, select **SQL Worksheet** to establish a connection and then open a worksheet for that connection.

Figure 5–7   Selecting the SQL Worksheet
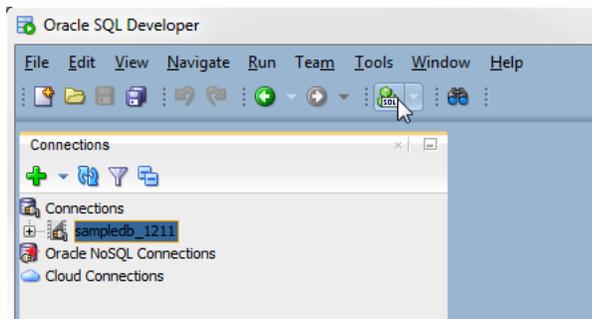
The **Select Connection** dialog appears. In the **Connection** field, select the name of the connection. Click **OK**.

Figure 5–8   Select Connection dialog
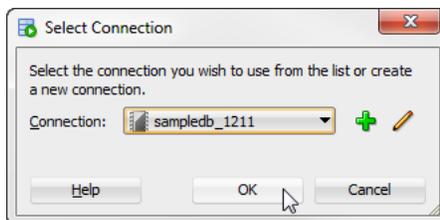
A connection to the TimesTen database is established if a connection associated with the selected connection name does not exist.

Before composing the statement in **SQL Worksheet**, ensure that appropriate indexes have been created on the tables that are being referenced in the statement, and that table and column statistics have been updated. For more information about updating statistics, see "Updating the table and column statistics used by the query optimizer" on page 5-1.

Compose the statement in the **SQL Worksheet**. Then choose **Explain Plan** to display the execution plan for the statement.

Figure 5–9   Viewing explain plan
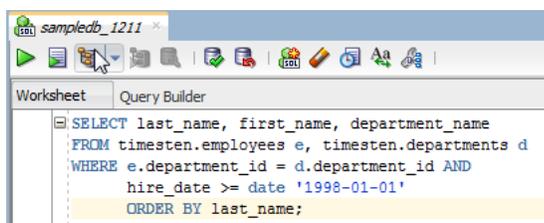
The execution plan is shown in the **Explain Plan** pane and is displayed in a hierarchical format in reverse chronological order such that the operation at the bottom of the tree is processed first and the operation at the top of the tree is processed last. For each operation, the plan shows the table or index the operation is performed on, or if applicable, the expressions of the statement's predicate the operation is processing. Types of operations that can appear in the execution plan include table or index scan (with lock granularity shown), join, group and sort.

Figure 5–10   Viewing the execution plan

For more information about interpreting an execution plan, see "Optimizer plan" in the *Oracle TimesTen In-Memory Database Introduction* or "Viewing SQL query plans" in the *Oracle TimesTen In-Memory Database Operations Guide*.

For more information about query performance tuning, see "Viewing SQL query plans" in the *Oracle TimesTen In-Memory Database Operations Guide*.

6

Working with the TimesTen Index Advisor

SQL Developer supports the TimesTen Index Advisor which can evaluate a SQL workload and recommend indexes. The indexes that the TimesTen Index Advisor recommends can improve the performance for the following: joins, single table scans, and `ORDER BY` or `GROUP BY` operations. The TimesTen Index Advisor can collect index information for the current connection or for the entire database.

> **Note:** SQL Developer uses procedures from the TimesTen Index Advisor to recommend indexes. For more information about the TimesTen Index Advisor, see "Using the Index Advisor to recommend indexes" in the *Oracle TimesTen In-Memory Database Operations Guide*.

Before you use the TimesTen Index Advisor:

■ Ensure that you are using Oracle TimesTen In-Memory Database Release 11.2.2.4 (or later).

This chapter provides information about working with the TimesTen index advisor.

Topics include:

■ Connection level capture

■ Database level capture

■ Saving index recommendations

■ Viewing the TimesTen index advisor capture state

Connection level capture

Before attempting to capture data for the TimesTen Index Advisor, ensure that your table statistics have been updated in the past 24 hours. Up-to-date table statistics provide the most up-to-date statistics for data collection and allow statements to be prepared with the optimized query plan. For more information about updating table statistics, see "Updating the table and column statistics used by the query optimizer" on page 5-1.

There are two ways to capture data for the TimesTen Index Advisor at the connection level.

■ SQL worksheet

■ SQL workload script

SQL worksheet

You can capture data for the TimesTen Index Advisor at the connection level from the SQL Developer SQL worksheet. Ensure that your SQL workload is in the SQL Developer SQL worksheet.

1. Click the **TimesTen Index Advisor** button from the SQL worksheet menu bar.

Figure 6–1 TimesTen Index Advisor from SQL worksheet
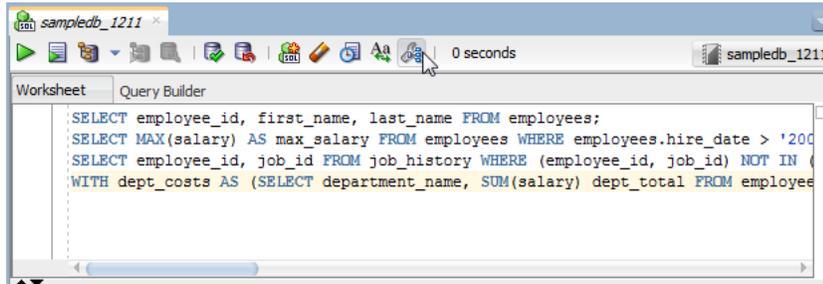
- If your table statistics are outdated, an information dialog displays information about the outdated tables. TimesTen recommends that you update your table statistics.

- If your table statistics are up to date, the **Index Advisor Configuration** dialog displays.

2. To update your table statistics, click **Cancel** and follow the steps in "Updating the table and column statistics used by the query optimizer" on page 5-1. To proceed without updating your table statistics, click **OK**.

Figure 6–2 Outdated table statistics
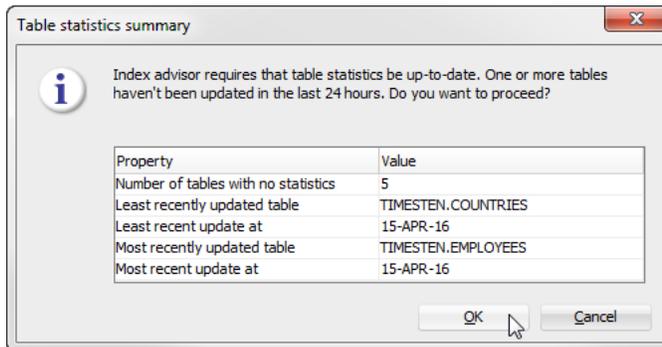
The **Index Advisor Configuration** dialog displays.

3. Select the desired capture mode from the **Capture mode** drop-down list:

- **Prepare SQL**: TimesTen prepares but does not execute the SQL commands from the SQL worksheet. The TimesTen Index Advisor makes recommendations using computed statistics and query plan analysis. This is the default capture mode.

- **Execute SQL**: TimesTen executes the SQL commands from the SQL worksheet. The TimesTen Index Advisor makes recommendations using the actual execution of the SQL commands. This capture mode may take longer than the Prepare SQL mode because TimesTen has to complete SQL execution.

Figure 6–3 Index Advisor configuration
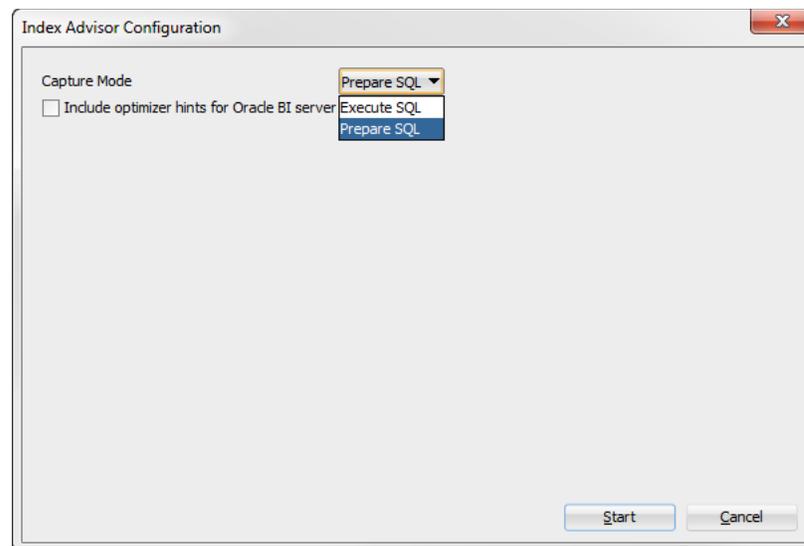
You can now choose to include optimizer hints for Oracle BI server.

4. The **Include optimizer hints for Oracle BI server** checkbox enables you to use optimizer hints that direct the TimesTen query optimizer to generate a specific execution plan. These optimizer hints are recommended queries generated by the Oracle BI server. For more information on using optimizer hints, see "Including optimizer hints for Oracle BI server in a SQL worksheet" on page 6-5. If you do not want to use optimizer hints, click **Start**.

Figure 6–4 Start collecting data for the TimesTen index advisor without optimizer hints
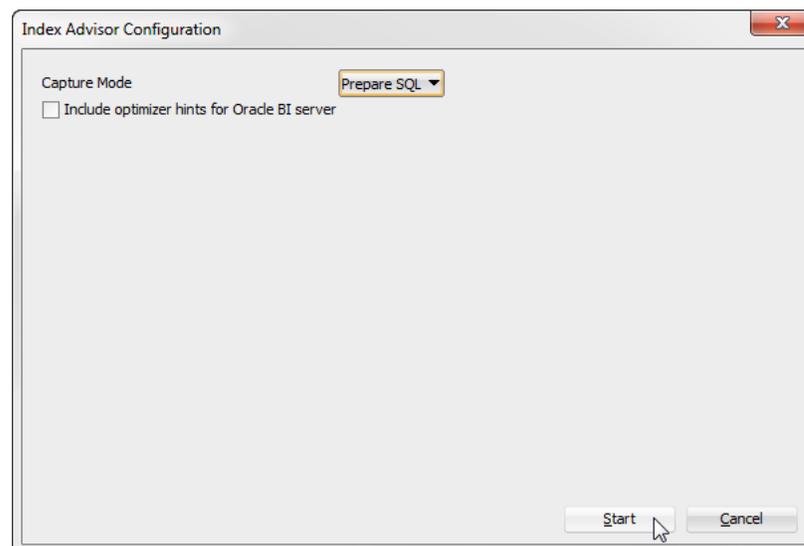
The **Index Advisor Configuration** dialog closes. A **TimesTen index advisor** pane displays at the bottom of the SQL worksheet.

5. In the **Selector** column of the TimesTen index advisor pane, select the index recommendations you want to create.

You can also click on the column header to sort the table based on the column. A description of each column follows:

- **Selector**

  The check box you select to determine the index recommendations that you want to create.

- **Index Recommendation**

  The CREATE INDEX statement that the TimesTen index advisor recommends.

- **Affected Statement Count**

  The number of statements that benefit from the recommendation.

- **Created**

  Shows if you have already created the index based on the recommendation by the TimesTen index advisor. This value can be Yes or No. If this value is Yes, TimesTen created the index successfully. If this value is No, TimesTen is in the process of creating the index or is unable to create the index.

To select all index recommendations, click the **Select all** check box.

To review the SQL commands that are being evaluated by the TimesTen index advisor, click the **SQL** tab. The **SQL** tab is located at the top right of the **TimesTen index advisor** pane.

6. Once you select the index recommendations that you want to create, click the **Create Selected Indexes** button.

Figure 6–5   Create index recommendations
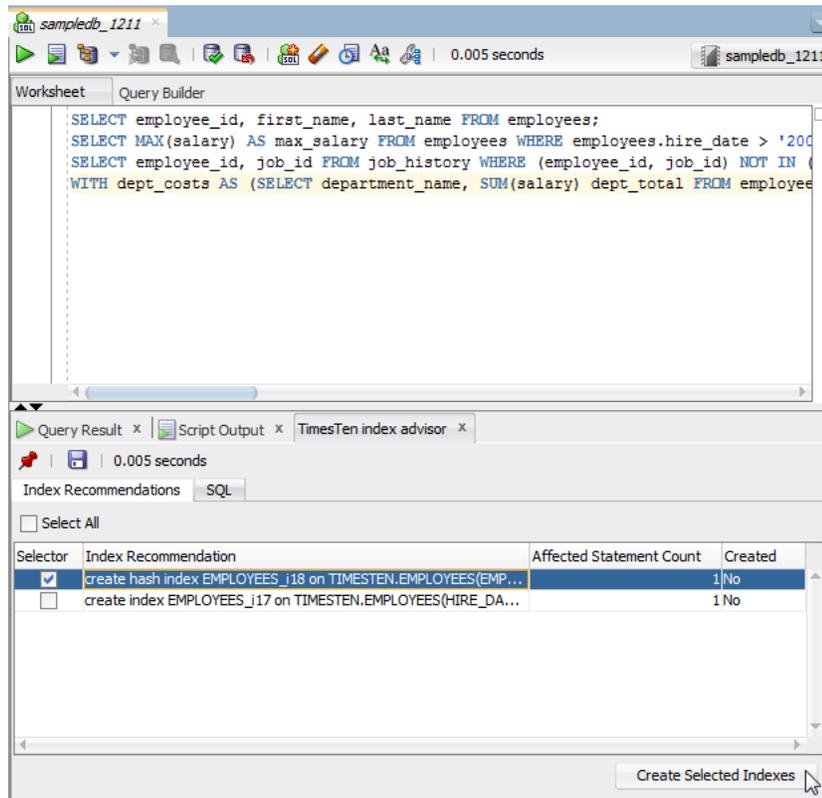
The **Create Selected Indexes** dialog displays. Locate the **Details >>** button.

7. Click **Details >>**.

Figure 6–6   Create selected indexes progress
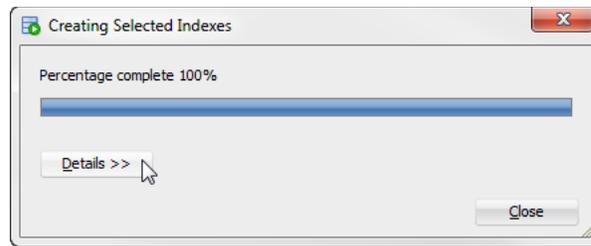
A details pane expands that shows progress information about creating the selected indexes. Locate the **Close** button at the bottom of the dialog.

8. Click **Close**.

Figure 6–7   Click close
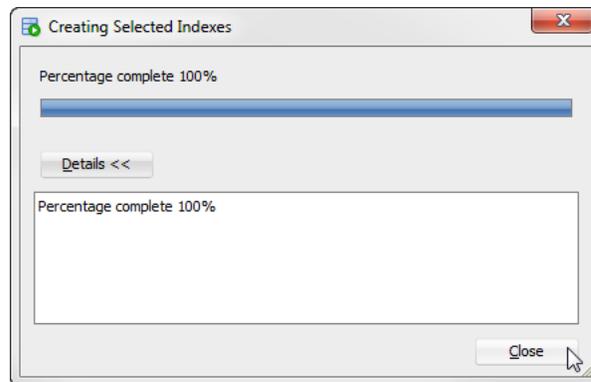
The **Create Selected Indexes** dialog closes. TimesTen creates the indexes. To save the index recommendations, see "Saving index recommendations from a SQL worksheet" on page 6-7.

Including optimizer hints for Oracle BI server in a SQL worksheet

Use transaction level optimizer hints to direct the TimesTen query optimizer to generate a specific execution plan.

> **Note:** For more information about optimizer hints, see "Use optimizer hints to modify the execution plan" in the *Oracle TimesTen In-Memory Database Operations Guide*.

Before using transaction level optimizer hints, ensure that you have completed the steps up to step 4 from "SQL worksheet" on page 6-2. When you use optimizer hints, ensure to not use any DDL statements or enable AUTOCOMMIT mode in your SQL workload.

To use optimizer hints for Oracle BI server, follow these steps:

**1.** Select the **Include optimizer hints for Oracle BI server** check box.

Figure 6–8   Including optimizer hints for Oracle BI server
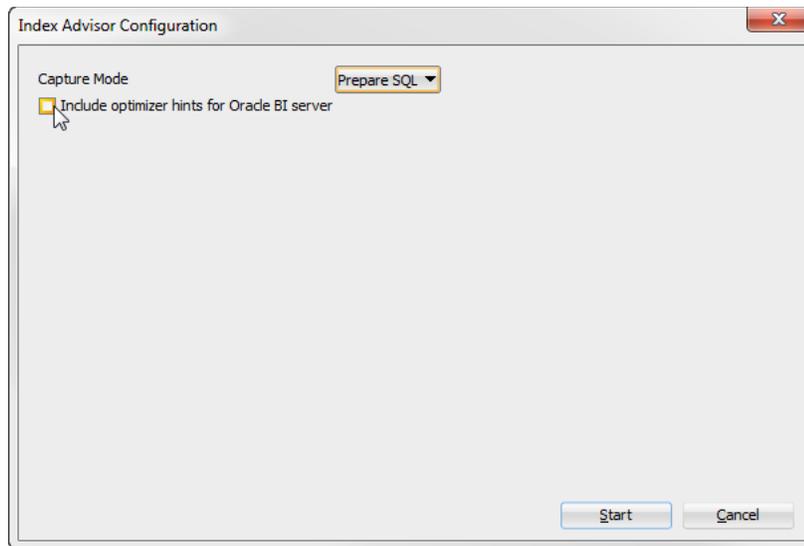
A field with the optimizer hints for Oracle BI server displays in the **Index Advisor Configuration** dialog. You cannot edit the optimizer hints. By default, the optimizer hints for Oracle BI server are as follows:

```
--
-- Optimizer hints to be used with TimesTen Index Advisor for Oracle BI server
--
autocommit 0;
call ttoptsetflag('rowlock',0);
call ttoptsetflag('tbllock',1);
call ttoptsetflag('mergejoin',0);
call ttoptsetflag('nestedloop',1);
```

Locate the **Start** button at the bottom right of the dialog.

2. Click **Start**.

Figure 6–9   Start collecting data for the TimesTen index advisor with optimizer hints
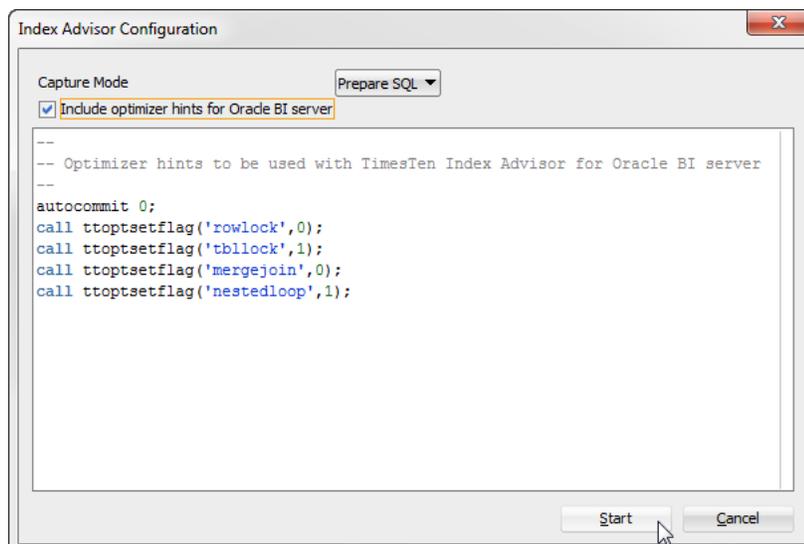
The **Index Advisor Configuration** dialog closes. A **TimesTen index advisor** pane displays at the bottom of the SQL worksheet. The AUTOCOMMIT mode of your transaction is returned to the AUTOCOMMIT mode that your transaction had prior to running the TimesTen index advisor.

For more information on the **TimesTen index advisor** pane and how to create indexes, see "In the Selector column of the TimesTen index advisor pane, select the index recommendations you want to create." on page 6-3 from "SQL worksheet" on page 6-2.

Saving index recommendations from a SQL worksheet

To save index recommendations for future reference, follow these steps:

1. Click **Save Indexes**. TimesTen saves all index recommendations. TimesTen does not save individual index recommendations.

Figure 6–10   Save Indexes
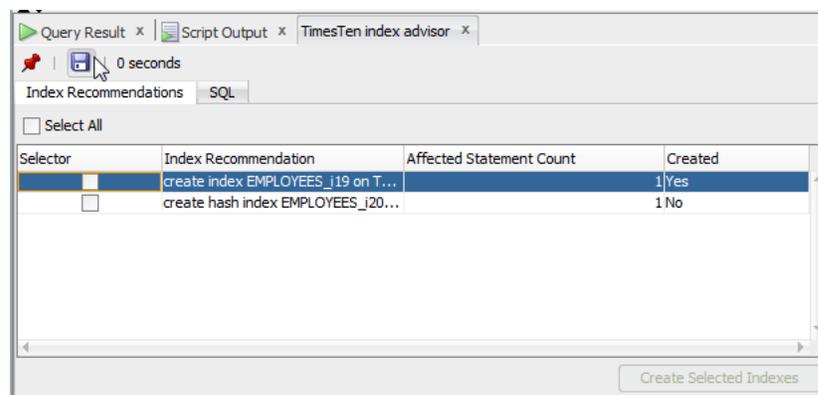
The **Save Indexes** dialog displays.

2. Select the directory where you want to save the TimesTen index advisor recommendations.

3. In the **File Name** field, define the file name of your TimesTen index advisor recommendations.

    The default filename of the TimesTen index advisor recommendations is *connection_name*-indexadvice-*YYYYMMDDMISS*.sql, where *connection_name* is the name of the connection. *YYYYMMDDHHMISS* is a timestamp of when the recommendations were made, where *YYYY* is the year, *MM* the month, *DD* the day, *HH* the hour, *MI* the minutes, and *SS* the seconds.

4. Click **Save**.

Figure 6–11   Save index recommendations
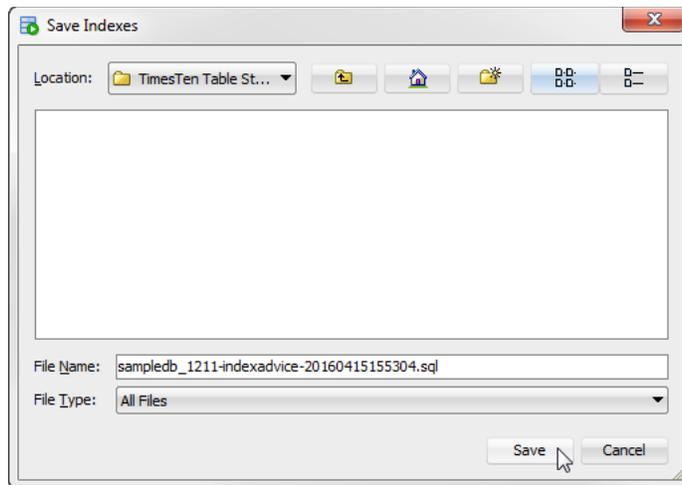
TimesTen saves the index recommendations.

The saved file header contains the following information:

- The timestamp of when TimesTen generated the report.

- The version of SQL Developer.

- The version of the TimesTen database.

- The name of the TimesTen connection.

For example:

```
-- This file was generated by SQL Developer at 2016-04-15 15:54:56
-- SQL Developer version 4.2.0.16.099.1111
-- Database version: 11.02.02.0008 Oracle TimesTen IMDB version 11.2.2.8.0
-- Connection name: sampledb_1211
```

SQL workload script

You can capture data for the TimesTen Index Advisor at the connection level using the SQL queries from a SQL workload script file. Ensure that you are on the main SQL Developer page and that your TimesTen connection is expanded.

1. Right-click the name of the database and select **Index Advisor**, then select **Connection Level Capture**.

Figure 6–12   Collecting data at the connection level
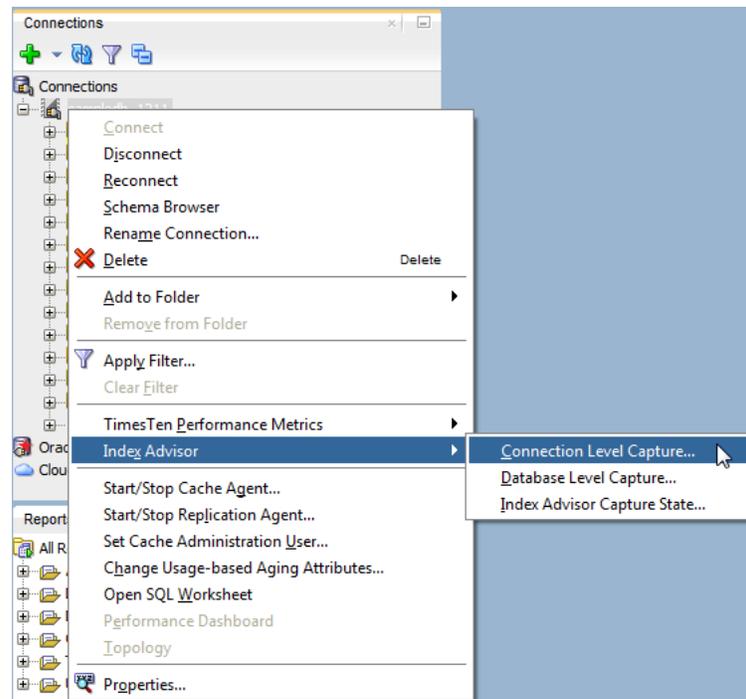
- If your table statistics are outdated, an information dialog displays information about the outdated tables. TimesTen recommends that you update your table statistics.

- If your table statistics are up to date, the **Index Advisor Configuration** dialog displays.

2. To update your table statistics, click **Cancel** and follow the steps in "Updating the table and column statistics used by the query optimizer" on page 5-1. To proceed without updating your table statistics, click **OK**.

*Figure 6–13   Outdated table statistics*
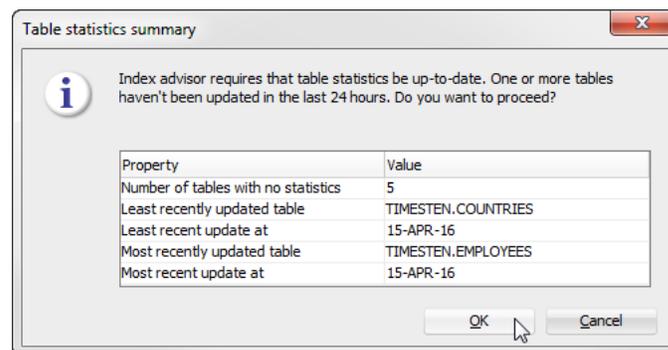
The **Index Advisor - Connection Level Capture** dialog displays.

3. In the **Configuration** tab of the **Index Advisor - Connect Level Capture** dialog, select the desired capture mode from the **Capture mode** drop-down list:

- **Prepare SQL**: TimesTen prepares but does not execute the SQL commands from the workload script. The TimesTen Index Advisor makes

recommendations using computed statistics and query plan analysis. This is the default capture mode.

■ **Execute SQL**: TimesTen executes SQL commands from the workload script. The TimesTen Index Advisor makes recommendations using the actual execution of the workload script. This capture mode may take longer than the Prepare SQL mode because TimesTen has to complete SQL execution.

Figure 6–14   Index advisor - Connection Level Capture
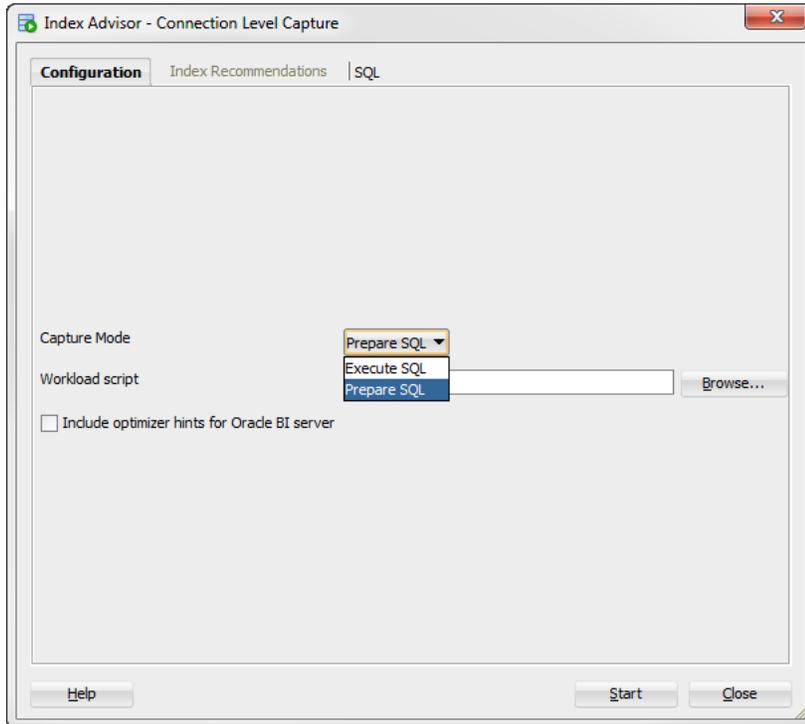
You are ready to specify your SQL workload script. SQL Developer uses the `ttIsql` utility to execute the SQL workload script. Locate the **Browse** button that is to the right of the **Workload script** field.

4. Click **Browse**.

Figure 6–15   Click Browse
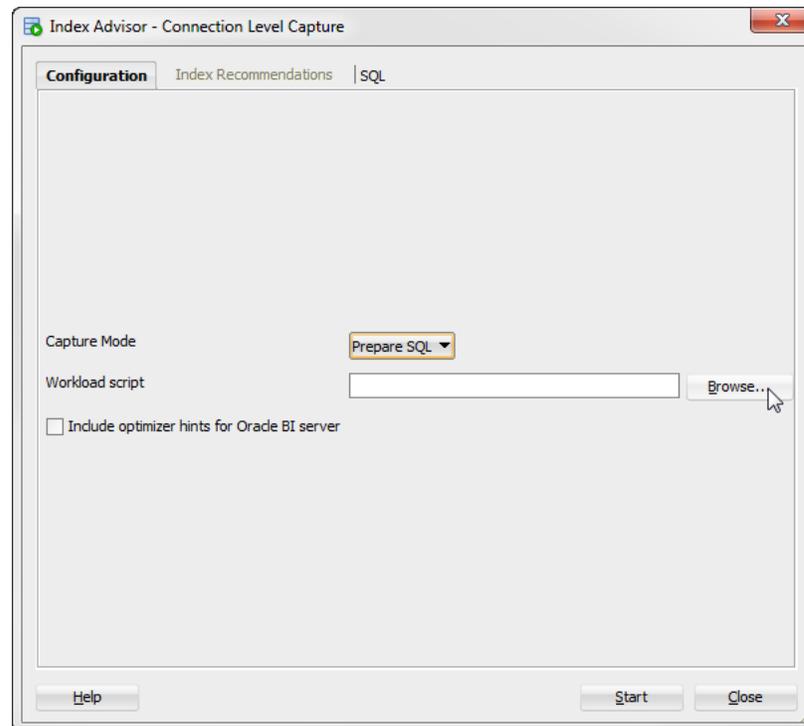
The **Open** dialog displays.

5. Select the directory where you saved your SQL workload script.

6. Select the SQL workload script for which you want to generate index recommendations.

7. Click **Open**.

Figure 6–16   SQL workload script open dialog
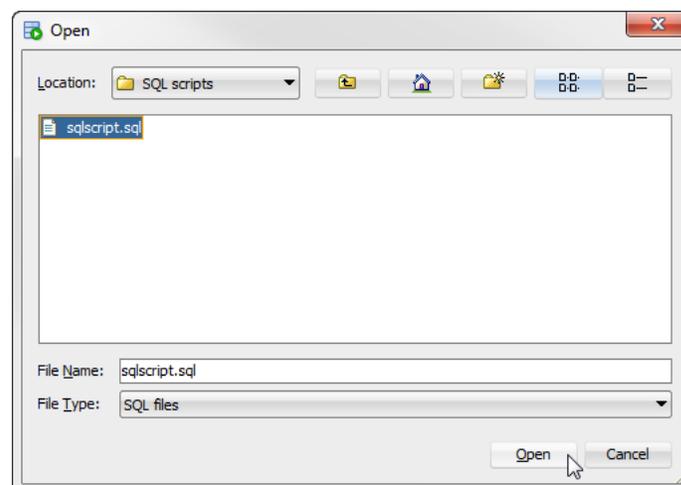
The TimesTen index advisor is now ready to collect data for your SQL workload. Locate the **Start** button.

To review the SQL commands that are being evaluated by the TimesTen index advisor, click the **SQL** tab. The **SQL** tab is located at the top of the **Index Advisor - Connection Level Capture** dialog.

8. The **Include optimizer hints for Oracle BI server** checkbox enables you to use optimizer hints that direct the TimesTen query optimizer to generate a specific execution plan. These optimizer hints are recommended queries generated by the Oracle BI server. For more information on using optimizer hints, see "Including optimizer hints for Oracle BI server for a SQL workload script" on page 6-15. If you do not want to use optimizer hints, click **Start**.

*Figure 6–17   Start collecting data for the TimesTen index advisor*
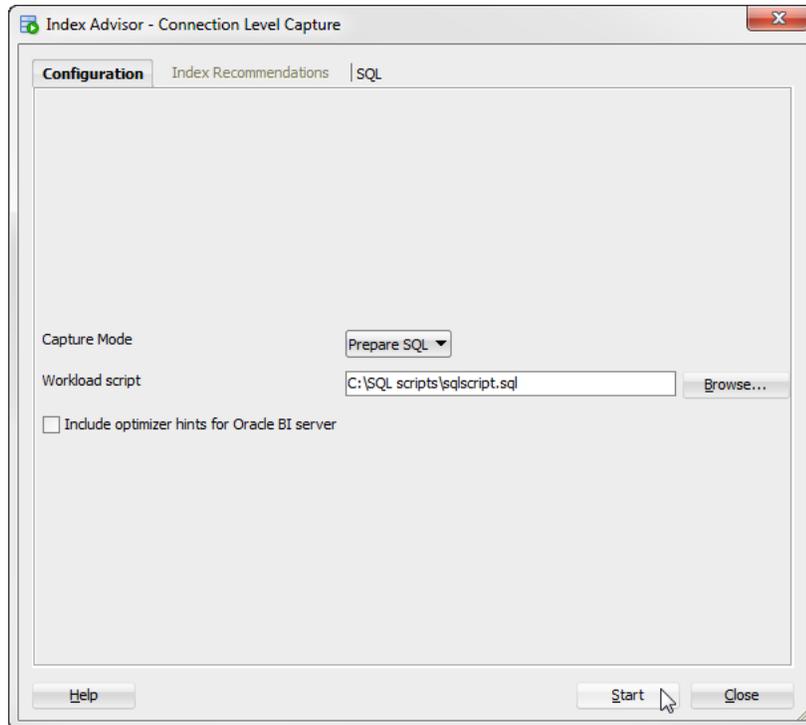
The **Index advisor assistant progress** dialog displays. Locate the **Details >>** button.

If the TimesTen index advisor does not have any recommendations, the **Index Recommendation Feedback** dialog displays. Close this dialog to return to the main SQL developer page.

9. Click **Details >>**.

Figure 6–18   Index advisor assistant progress
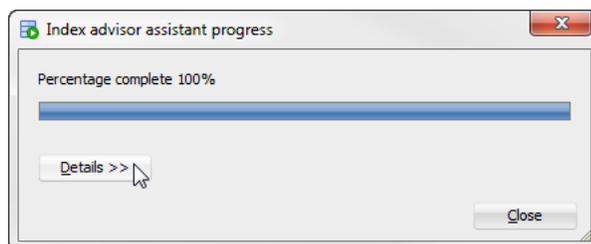
A details pane expands that shows progress information about the index advisor connection level capture. Locate the **Close** button at the bottom of the dialog.

10. Click **Close**.

Figure 6–19   Click Close
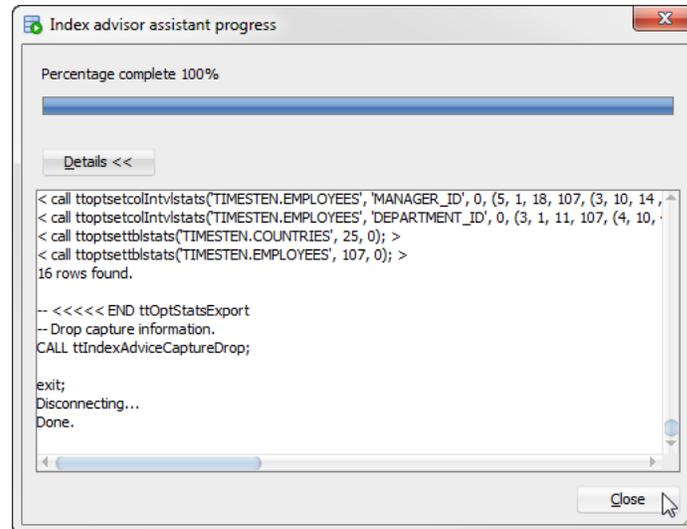
The **Index advisor assistant progress** dialog closes. Locate the **Index Recommendations** tab.

11. Click the **Index Recommendations** tab.

Figure 6–20   Index recommendations for connection level capture
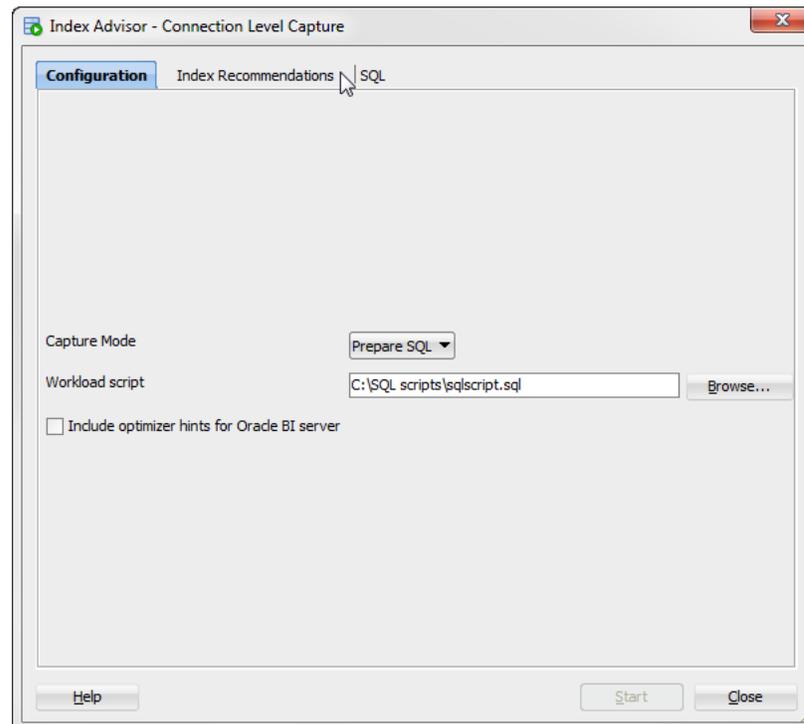
The **Index Recommendations** tab of the **Index Advisor - Connection Level Capture** dialog displays.

12. In the **Selector** column, select the index recommendations that you want to create.

You can also click on the column header to sort the table based on the column. A description of each column follows:

- **Selector**

  The check box you select to determine the index recommendations that you want to create.

- **Index Recommendation**

  The `CREATE INDEX` statement that the TimesTen index advisor recommends.

- **Affected Statement Count**

 The number of statements that benefit from the recommendation.

- **Created**

  Shows if you have already created the index based on the recommendation by the TimesTen index advisor. This value can be Yes or No. If this value is Yes, TimesTen created the index successfully. If this value is No, TimesTen is in the process of creating the index or is unable to create the index.

To select all index recommendations, click the **Select all** check box.

13. Once you select the index recommendations that you want to apply, click the **Create Selected Indexes** button.

Figure 6–21 Create index recommendations
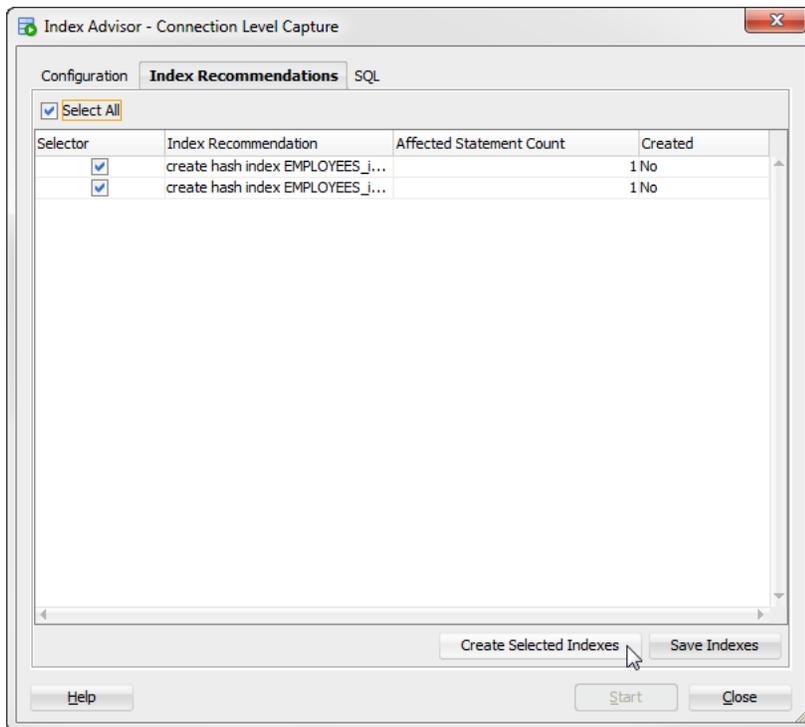
The **Creating Selected Indexes** dialog displays. Locate the **Details >>** button.

14. Click **Details >>**.

Figure 6–22   Create Selected Indexes
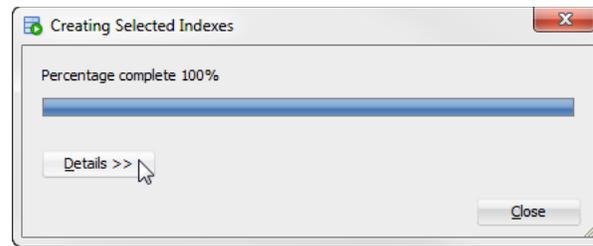
A details pane expands that shows progress information about creating the selected indexes. Locate the **Close** button at the bottom of the dialog.

15. Click **Close**.

Figure 6–23   Click close
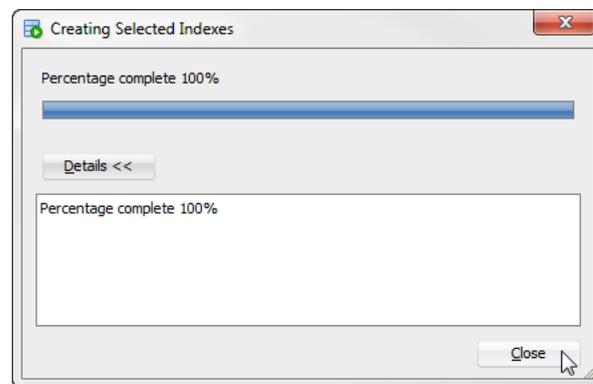
The **Creating Selected Indexes** dialog closes. The indexes are created. To save the index recommendations, see "Saving index recommendations" on page 6-27.

Including optimizer hints for Oracle BI server for a SQL workload script

Use transaction level optimizer hints to direct the TimesTen query optimizer to generate a specific execution plan.

> **Note:** For more information about transaction level optimizer hints, see "Use optimizer hints to modify the execution plan" in the *Oracle TimesTen In-Memory Database Operations Guide*.

Before using transaction level optimizer hints, ensure you have completed the steps up to step 8 from "SQL workload script" on page 6-8. When you use optimizer hints, ensure to not use any DDL statements or enable AUTOCOMMIT mode in your SQL workload.

To use optimizer hints for Oracle BI server, follow these steps:

**1.** Select the **Include optimizer hints for Oracle BI server** check box.

Figure 6–24   Including optimizer hints for Oracle BI server
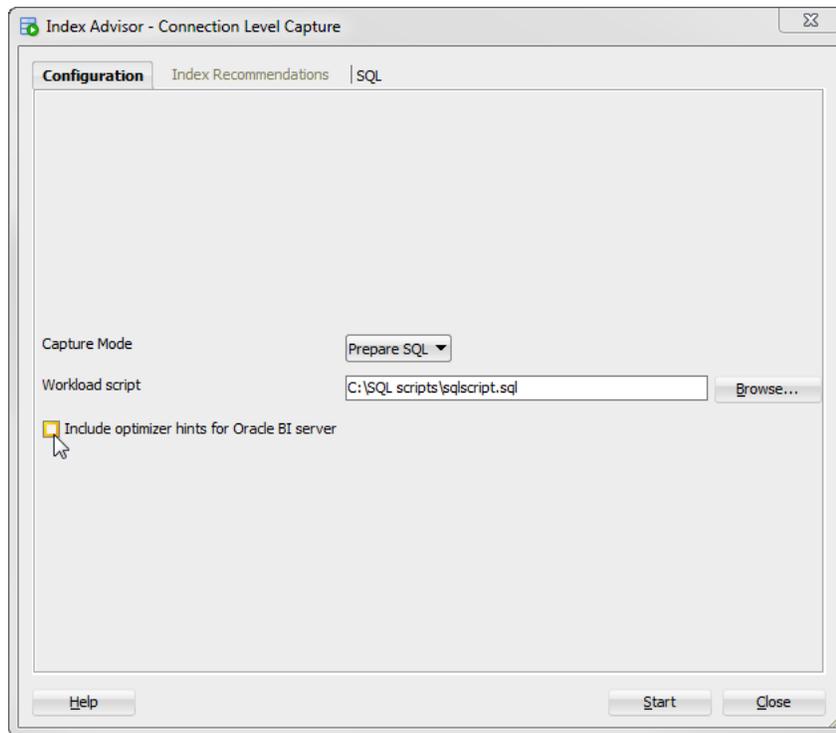
A field with the optimizer hints for Oracle BI server displays in the **Index Advisor - Connection Level Capture** dialog. These optimizer hints are not editable. By default, the optimizer hints for Oracle BI server are as follows:

```
--
-- Optimizer hints to be used with TimesTen Index Advisor for Oracle BI server
--
autocommit 0;
call ttoptsetflag('rowlock',0);
call ttoptsetflag('tbllock',1);
call ttoptsetflag('mergejoin',0);
call ttoptsetflag('nestedloop',1);
```

Locate the **Start** button at the bottom right of the dialog.

2.   Click **Start**.

*Figure 6–25   Start collecting data for the TimesTen index advisor with optimizer hints*
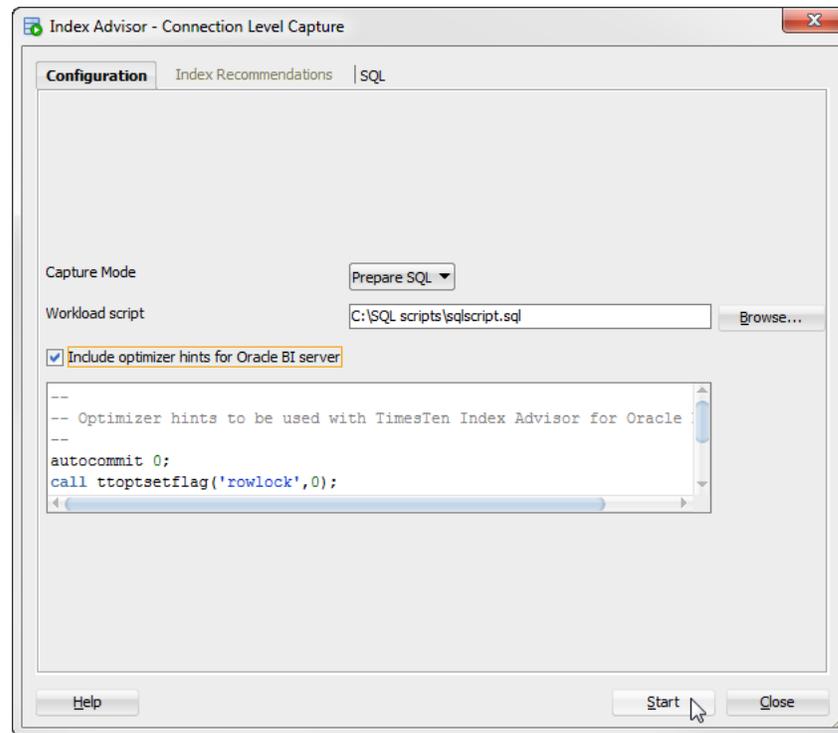
The **Index advisor assistant progress** dialog displays. Locate the **Details >>** button.

3. Click **Details >>**.

*Figure 6–26   Index advisor assistant progress*
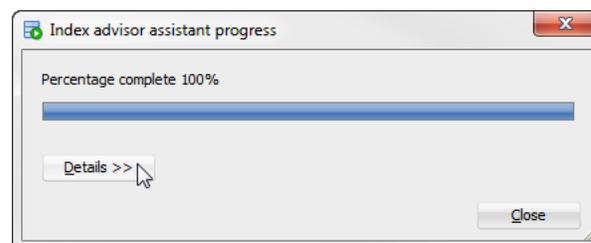
A details pane expands that shows progress information about the index advisor connection level capture. Locate the **Close** button at the bottom of the dialog.

4. Click **Close**.

Figure 6–27   Click Close
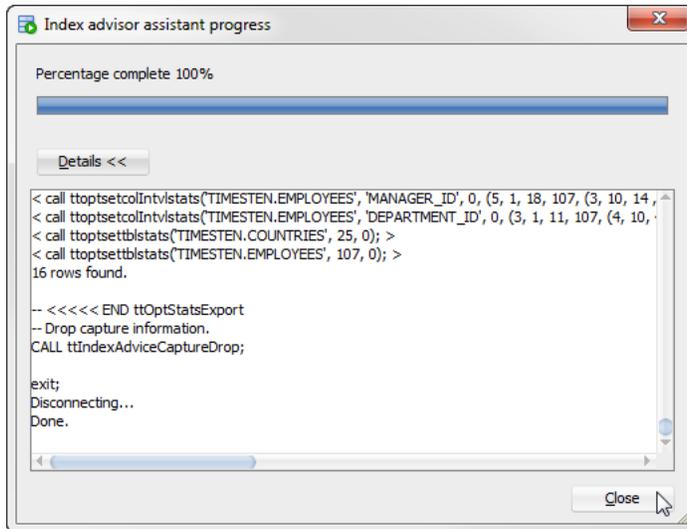
The **Index advisor assistant progress** dialog closes. Locate the **Index Recommendations** tab.

5.   Click the **Index Recommendations** tab.

Figure 6–28   Index recommendations for connection level capture
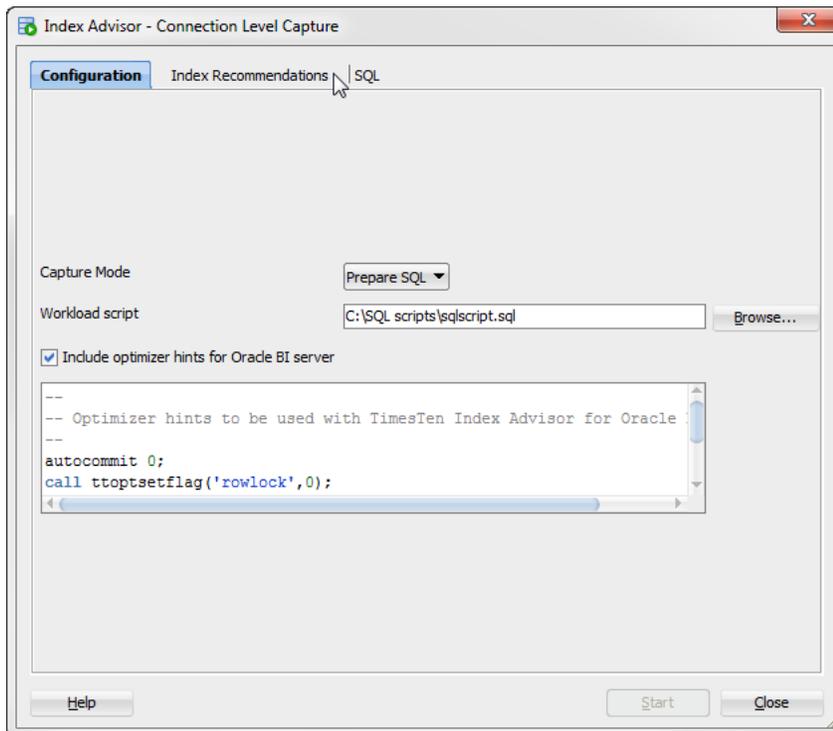
The **Index Recommendations** tab of the **Index Advisor - Connection Level Capture** dialog displays.

6.   In the **Selector** column, select the index recommendations that you want to create.

You can also click on the column header to sort the table based on the column. A description of each column follows:

- **Selector**

  The check box you select to determine the index recommendations that you want to create.

- **Index Recommendation**

  The `CREATE INDEX` statement that the TimesTen index advisor recommends.

- **Affected Statement Count**

 The number of statements that benefit from the recommendation.

- **Created**

  Shows if you have already created the index recommendation. This value can be Yes or No.

 To select all index recommendations, click the **Select all** check box.

7. Once you select the index recommendations that you want to apply, click the **Create Selected Indexes** button.

Figure 6–29   Create index recommendations
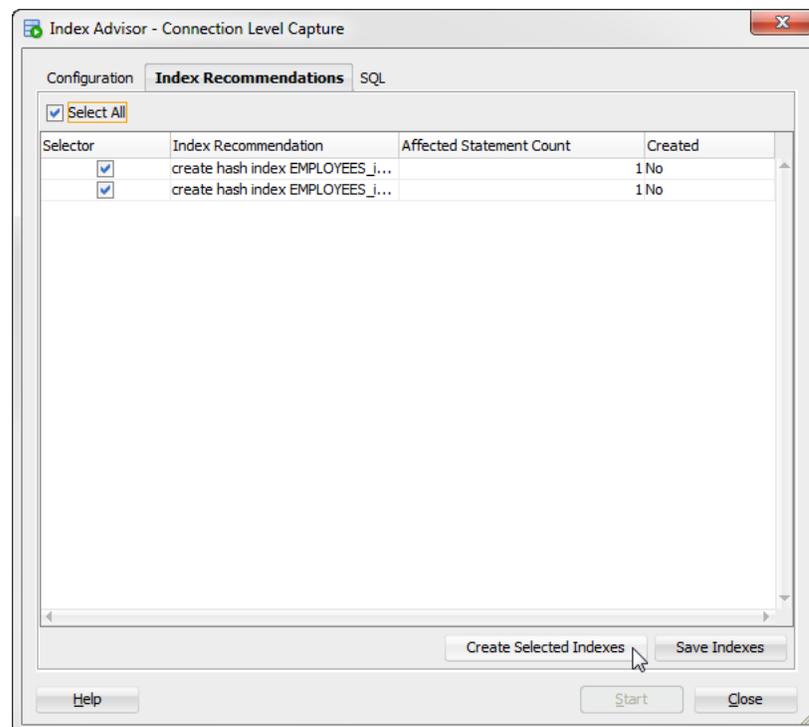
The **Creating Selected Indexes** dialog displays. Locate the **Details >>** button.

8. Click **Details >>**.

Figure 6–30   Creating Selected Indexes
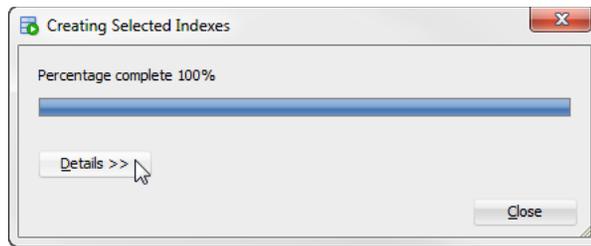
A details pane expands that shows progress information about creating the selected indexes. Locate the **Close** button at the bottom of the dialog.

9. Click **Close**.

Figure 6–31   Click close
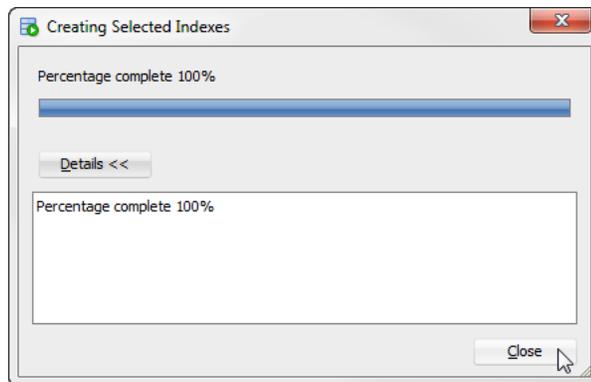
The **Creating Selected Indexes** dialog closes. The indexes are created. The AUTOCOMMIT mode of your transaction is returned to the AUTOCOMMIT mode that your transaction had prior to running the TimesTen index advisor. To save the index recommendations, see "Saving index recommendations" on page 6-27.

Database level capture

Before attempting to capture data for the TimesTen Index Advisor, ensure that your table statistics have been updated in the past 24 hours. Up-to-date table statistics provide the most up-to-date statistics for data collection and allow statements to be prepared with the optimized query plan.For more information about updating table statistics, see "Updating the table and column statistics used by the query optimizer" on page 5-1.

Also, ensure that you have granted ADMIN privileges to the TimesTen connection user. For more information on the GRANT SQL statement, see "GRANT" in the *Oracle TimesTen In-Memory Database SQL Reference*.

To collect data for the TimesTen index advisor at the database level, ensure that you are on the main SQL Developer page and that your TimesTen connection is expanded.

1. Right-click the name of the database and select **Index Advisor**, then select **Database Level Capture**.

Figure 6–32   Collecting data at the database level
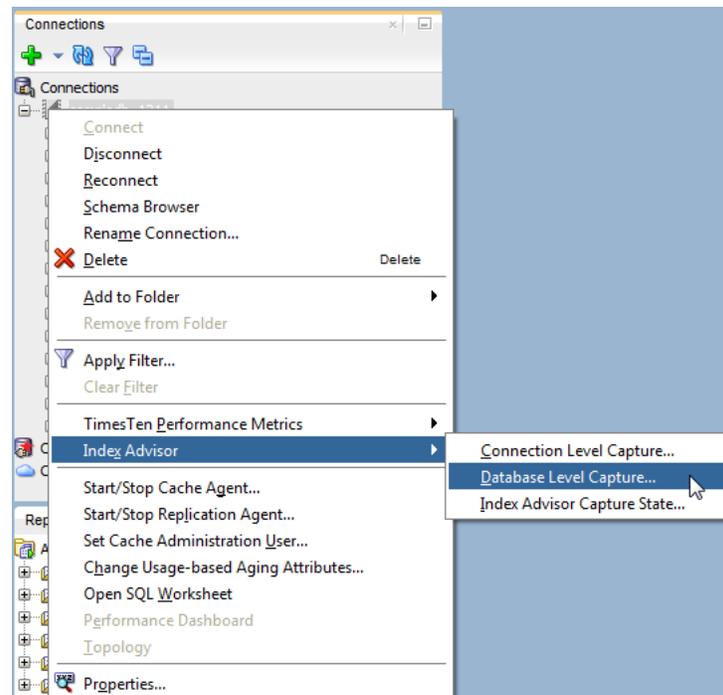
- If your table statistics are outdated, an information dialog displays information about the outdated tables. It is recommended that your table statistics are up to date.

- If your table statistics are up to date, the **Index Advisor - Connection Level Capture** dialog displays.

2. To update your table statistics, click **Cancel** and follow the steps in "Updating the table and column statistics used by the query optimizer" on page 5-1. To proceed without updating your table statistics, click **OK**.

*Figure 6–33   Outdated table statistics*
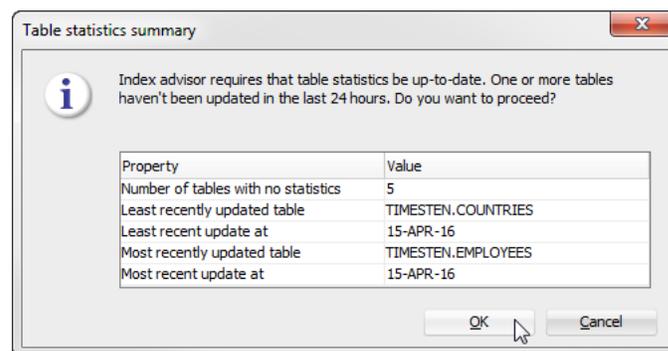
The **Index Advisor - Database Capture** dialog displays. Locate the **Start** button at the bottom of the dialog.

3. In the **Control** tab of the **Index Advisor - Connect Level Capture** dialog, click **Start**.

Figure 6–34   Index Advisor - Database Capture
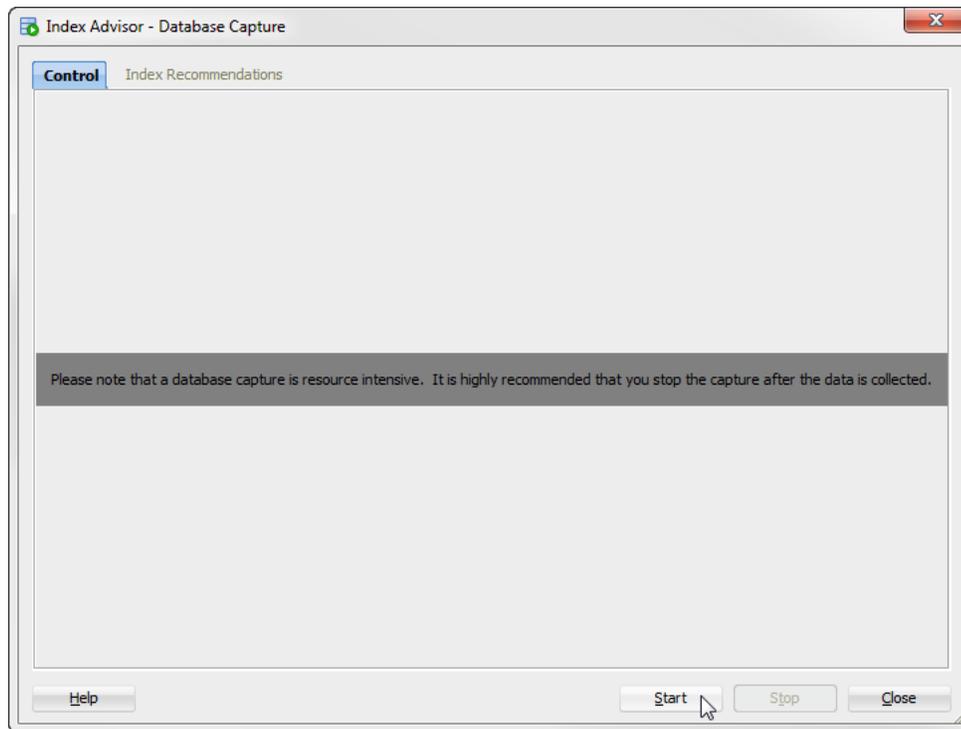

The **Index Advisor Database Capture In Progress** dialog displays. Locate the **Details >>** button.

4.   Click **Details >>**.

Figure 6–35   Index advisor database capture progress
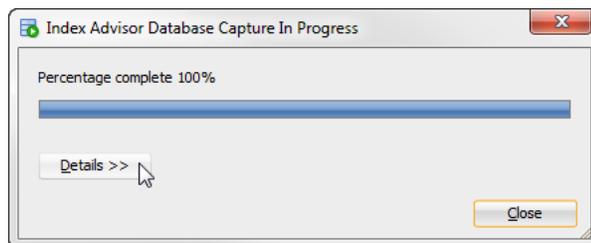
A details pane expands that shows progress information about the index advisor database level capture. Locate the **Close** button at the bottom of the dialog.

5.   Click **Close**.

Figure 6–36   Click Close
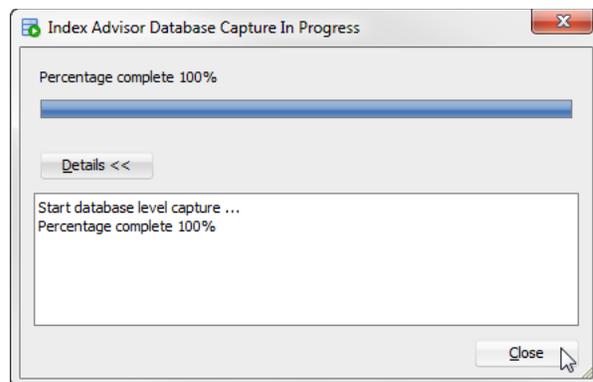
The **Index Advisor Database Capture In Progress** dialog closes.

6. Once you have captured your desired SQL workload, click **Stop**.

Figure 6–37   Stop database capture
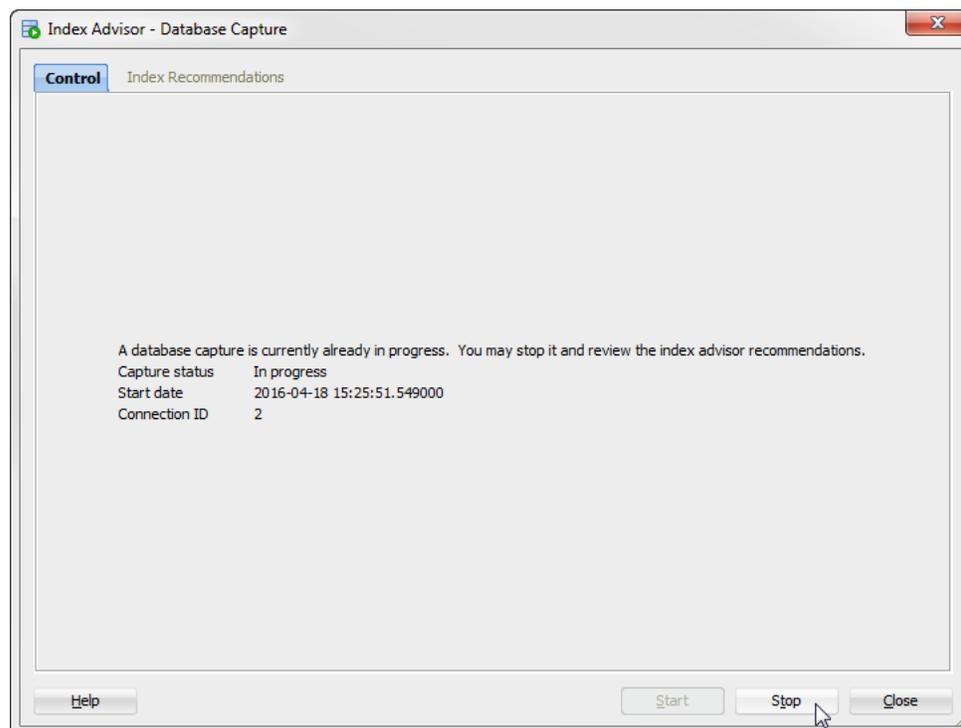
If the TimesTen index advisor has recommendations, the **Index Advisor Database Capture In Progress** dialog displays.

If the TimesTen index advisor does not have any recommendations, the **Index Recommendation Feedback** dialog displays. Close this dialog to return to the main SQL developer page.

7. Click **Details >>**.

Figure 6–38   Index advisor database capture progress
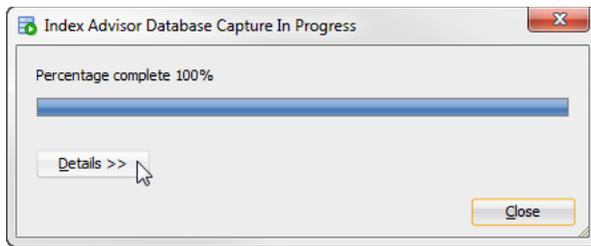
A details pane expands that shows progress information about the index advisor database capture. Locate the **Close** button at the bottom of the dialog.

8. Click **Close**.

Figure 6–39   Click Close
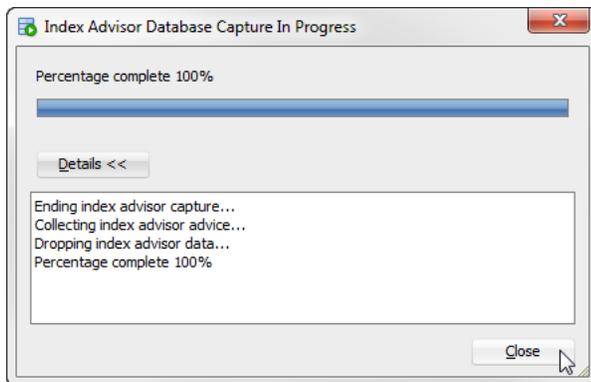
The **Index Advisor Database Capture In Progress** dialog closes. The Index Advisor Database Capture dialog shows information about the completed database capture. Locate the **Index Recommendations** tab at the top of the dialog.

9. Click the **Index Recommendation**s tab.

Figure 6–40 Index recommendations for database capture
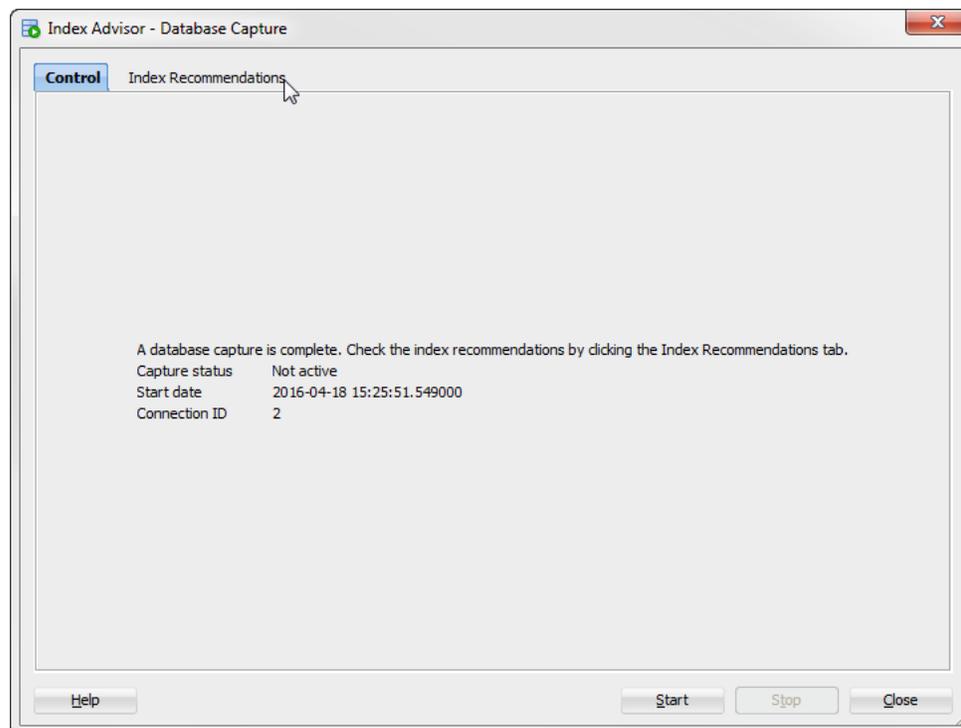
The **Index Recommendations** tab of the **Index Advisor - Database Capture** dialog displays.

10. In the **Selector** column, select the index recommendations that you want to create.

You can also click on the column header to sort the table based on the column. A description of each column follows:

- **Selector**

  The check box you select to determine the index recommendations that you want to create.

- **Index Recommendation**

  The CREATE INDEX statement that the TimesTen index advisor recommends.

- **Affected Statement Count**

  The number of statements that benefit from the recommendation.

- **Created**

  Shows if you have already created the index recommendation. This value can be Yes or No.

To select all index recommendations, click the **Select all** check box.

11. Once you select the index recommendations that you want to apply, click the **Create Selected Indexes** button.

Figure 6–41   Create index recommendations
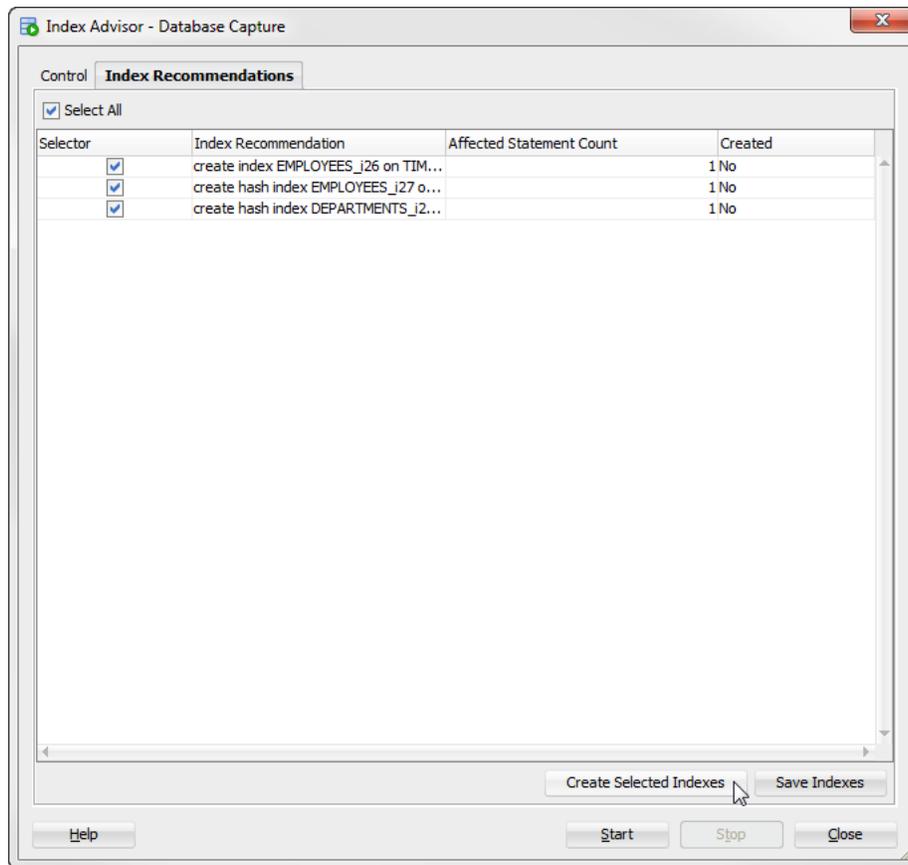
The **Creating Selected Indexes** dialog displays. Locate the **Details >>** button.

12. Click **Details >>**.

Figure 6–42   Creating Selected Indexes
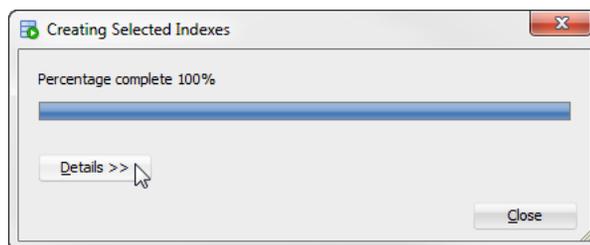
A details pane expands that shows progress information about creating the selected indexes. Locate the **Close** button at the bottom of the dialog.

13. Click **Close**.

Figure 6–43   Click close
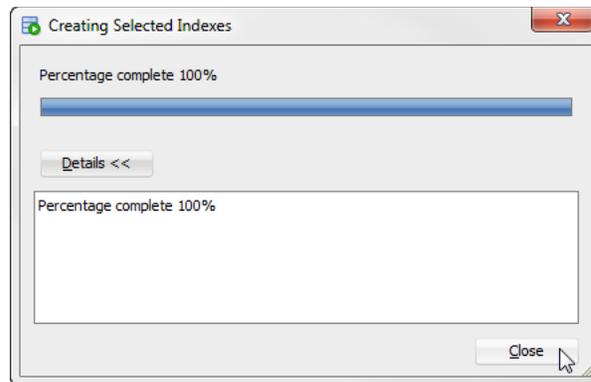
The **Creating Selected Indexes** dialog closes. The indexes are created. To save the index recommendations, see "Saving index recommendations" on page 6-27.

Saving index recommendations

These steps describe how to save index recommendations for future reference. These steps are valid for "Database level capture" on page 6-20 for a database level capture and "SQL workload script" on page 6-8 for a connection level capture.

1. Click **Save Indexes**. By default, all index recommendations are saved. You cannot save individual index recommendations.

Figure 6–44   Save Indexes
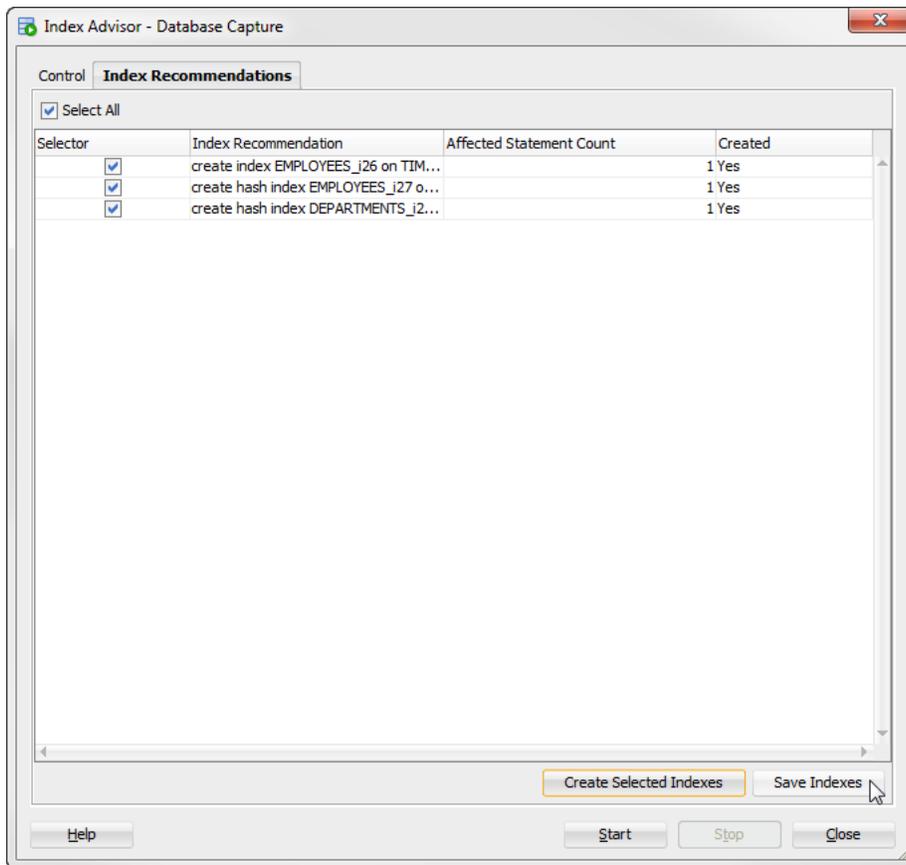
The **Save Indexes** dialog displays.

2. Select the directory where you want to save the TimesTen index advisor recommendations.

3. In the **File Name** field, define the file name of your TimesTen index advisor recommendations.

   The default filename of the TimesTen index advisor recommendations is `connection_name`-indexadvice-`YYYYMMDDMISS`.sql, where `connection_name` is the name of the connection. `YYYYMMDDHHMISS` is a timestamp of when the recommendations were made, where `YYYY` is the year, `MM` the month, `DD` the day, `HH` the hour, `MI` the minutes, and `SS` the seconds.

4. Click **Save**.

Figure 6–45   Save index recommendations
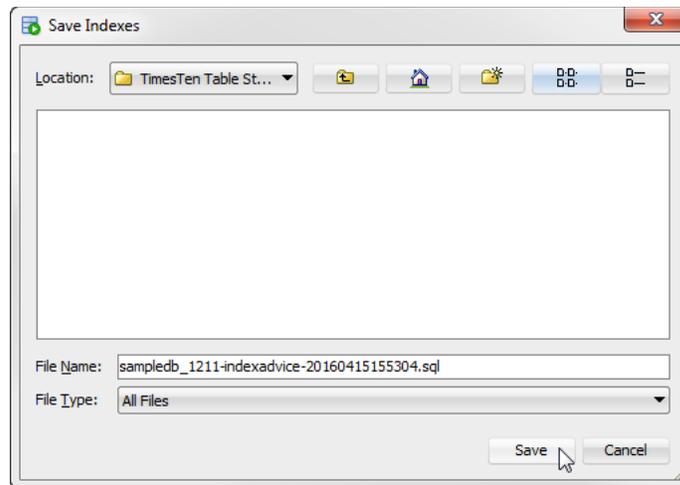
Your index recommendations are saved.

The saved file contains a header with following information:

- The timestamp of when the report was generated.

- The version of SQL Developer.

- The version of the TimesTen database.

- The name of the TimesTen connection.

For example:

```
-- This file was generated by SQL Developer at 2016-04-18 15:54:56
-- SQL Developer version 4.2.0.16.099.1111
-- Database version: 11.02.02.0008 Oracle TimesTen IMDB version 11.2.2.8.0
-- Connection name: sampledb_1211
```

Viewing the TimesTen index advisor capture state

The TimesTen index advisor capture state enables you to view the status of connection and database level captures. To view the TimesTen index advisor capture state for a database, ensure that you are on the main SQL Developer page.

1. Right-click the name of the database and select **Index Advisor**, then select **Index Advisor Capture State**.

Figure 6–46   Viewing the TimesTen index advisor capture state
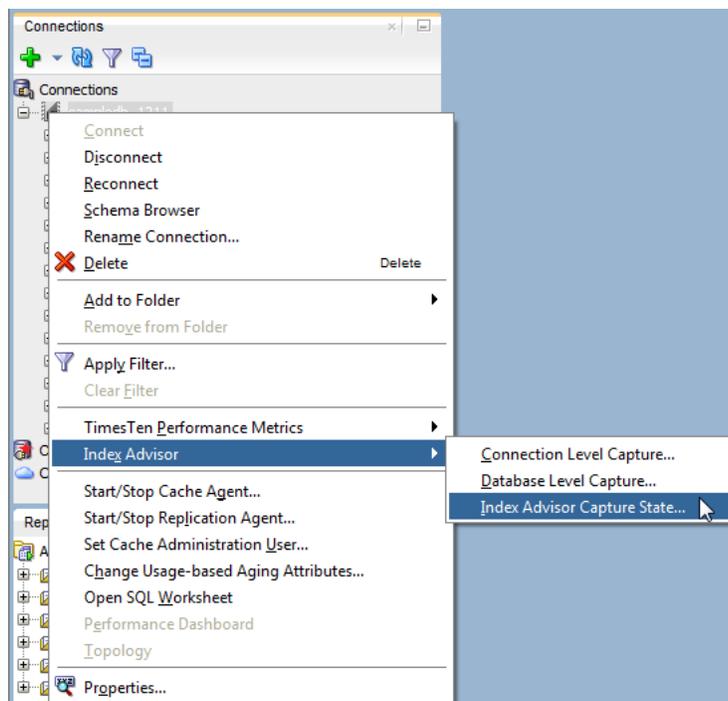
The **Index Advisor Capture State** dialog displays and you see the current status of connection and database level captures.

You can also click on the column header to sort the table based on the column. A description of each column follows:

- **Connection ID**

  The connection ID of the connection that initiated the TimesTen index advisor capture.

- **Capture Level**

  The capture level of the TimesTen index advisor capture. This value can be Connection or Database.

- **Capture Mode**

  The capture mode of the TimesTen index advisor capture. This value can be Execute SQL or Prepare SQL.

- **Capture State**

  The capture state of the TimesTen index advisor capture. This value can be In Progress or Completed.

- **Prepared Count**

  The number of prepared SQL statements during the capture period.

- **Execute Count**

  The number of executed SQL statements during the capture period.

- **Start Time**

The timestamp of when the index advisor capture began. The timestamp format is *YYYY-MM-DD HH:MI:SS.FF*, where *YYYY* is the year, *MM* the month, *DD* the day, *HH* the hour, *MI* the minutes, *SS* the seconds, and *FF* the milliseconds.

■ **End Time**

The timestamp of when the index advisor capture ended. The timestamp format is *YYYY-MM-DD HH:MI:SS.FF*, where *YYYY* is the year, *MM* the month, *DD* the day, *HH* the hour, *MI* the minutes, *SS* the seconds, and *FF* the milliseconds. This value is empty if the **Capture State** is in In Progress.

2. Click **Refresh** to update the table with the most current information of the TimesTen index advisor captures.

Figure 6–47 Click refresh
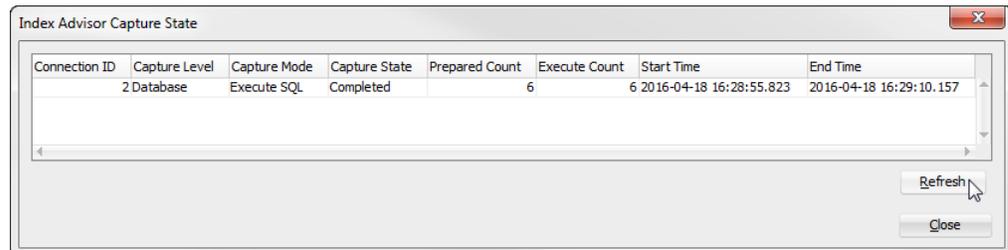
The **Index Advisor Capture State** dialog is refreshed with the most recent capture state information.

7

Loading data from an Oracle database into a TimesTen database

SQL Developer allows you to load data using parallel threads from an Oracle database into a TimesTen database without creating a cache group or cache table.

> **Note:** SQL Developer uses functions from the `ttLoadFromOracle` built-in procedure to load data from an Oracle database into a TimesTen database. For more information about the `ttLoadFromOracle` utility, see "ttLoadFromOracle" in the *Oracle TimesTen In-Memory Database Reference*.

This chapter provides information about loading Oracle database data into a TimesTen table.

Topics include:

- Loading data from Oracle database tables
- Loading data from an Oracle database table into an existing TimesTen table
- Resume loading data from Oracle database tables

Loading data from Oracle database tables

Before attempting to load data from an Oracle database into a TimesTen database:

- Ensure that you are using Oracle TimesTen In-Memory Database Release 11.2.2.4 (or later).
- Ensure that you have specified the Oracle database connection information (Oracle Database Password and Oracle Net Service Name) for your TimesTen SQL developer connection. For more information on defining your TimesTen database connection, see "Defining a TimesTen database named connection" on page 3-1.

To load data from an Oracle database into a TimesTen database, ensure that you are on the main SQL Developer page and that your TimesTen connection is expanded.

1. Right-click the **Tables** option and select **Load from Oracle Database Tables**.

Figure 7–1   Loading data from Oracle database tables
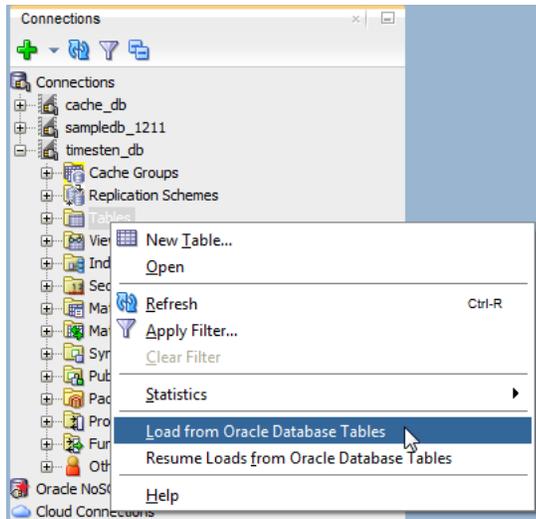
The **Load from Oracle Database Tables** dialog displays. Locate your Oracle database schema name from the navigation tree on the left of the **Load from Oracle Database Tables** dialog.

2. Select a mode:

   ■   Simple - This option enables you to load data from an Oracle database.

   ■   Advanced - This option enables you to load data from an Oracle database and specify the mapping of Oracle database data types to TimesTen database data types. You can also specify additional load options.

3. Once you have selected a mode, click **Next>**.

Figure 7–2   Click Next>
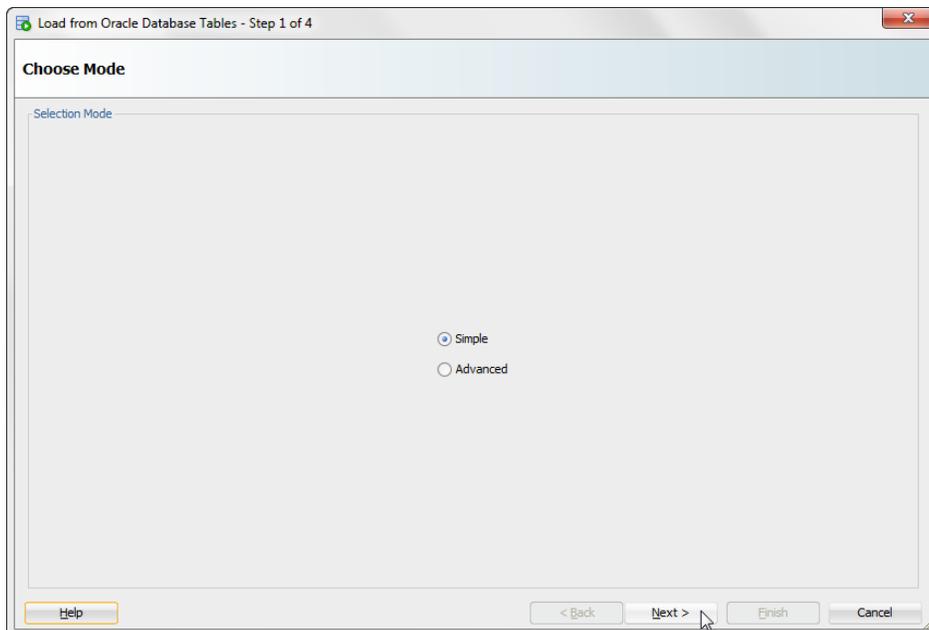
Depending on the mode that you selected, follow the instructions from "Advanced" on page 7-3 or "Simple" on page 7-7.

Advanced

The advanced mode enables you to select the mapping of Oracle database data types to TimesTen data types and get data compression recommendations through compression analysis. Before using the advanced mode, ensure that you have completed all the steps from "Loading data from Oracle database tables" on page 7-1.

> **Note:** The data type mapping and compression settings set in this section are applied to all the tables that you load from the Oracle database.

> **Note:** You can click the **Reset Factory Defaults** button to set all configuration values to factory defaults.

---

1. In the Type Mapping section of the Type Mapping & Compression tab, choose the data type mapping for the following data types:

   - **NUMBER**: Data type mapping for the NUMBER data type. You can select the following data type mappings:

     - **No Mapping**: Columns of data type NUMBER are not analyzed and are imported with the same data type as in the Oracle database table.

     - **Standard Mapping**: Columns of data type NUMBER with precision zero are mapped to a suitable TimesTen native integer data type. For more information on native integer data types, see "Numeric data types" in the *Oracle TimesTen In-Memory Database SQL Reference*.

     - **Aggressive Mapping**: All columns of data type NUMBER are mapped in the TimesTen database as columns of data type NUMBER with the maximum precision and scale used in the Oracle database table.

   - **VARCHAR2**: Data type mapping for the VARCHAR2 and NVARCHAR2 data type. You can select the following data type mappings:

     - **No Mapping**: Columns of data type VARCHAR2 and NVARCHAR2 are not analyzed and are imported with the same data type as in the Oracle database table.

     - **Aggressive Mapping**: All columns of data type VARCHAR2 and NVARCHAR2 are mapped in the TimesTen database as columns with the maximum precision and scale used in the Oracle database table.

   - **TIMESTAMP with TIME ZONE**: Data type mapping for the TIMESTAMP WITH TIMEZONE data type. You can select the following data mapping types:

     - Treat as unsupported type: Columns of data type TIMESTAMP WITH TIMEZONE are treated as an unsupported data type. If you select Stop on Any Bad Type as the Analysis Parameter, the data type mapping analysis stops when it encounters a TIMESTAMP WITH TIMEZONE data type column. If you select any other option for the Analysis Parameter, the data type mapping analysis continues and TIMESTAMP WITH TIMEZONE data type columns are ignored during table loading.

–   **Map to TIMESTAMP**: Columns of data type `TIMESTAMP WITH TIMEZONE` are mapped in the TimesTen database as `TIMESTAMP` data type columns. However, all the timezone information is lost because the `TIMESTAMP` data type does not support timezone information.

■   **LOB data types**: Data type mapping for any LOB data type. You can select the following data mapping types:

–   Map to VAR*: This maps columns of LOB data types from the Oracle database to VAR data types in the TimesTen database:

`BLOB` as `VARBINARY(4194304)`

`CLOB` as `VARCHAR2(4194304)`

`NCLOB` as `NVARCHAR2(2097152)`

■   **RAW data types**: Data type mapping for any RAW data type. You can select the following data mapping types:

–   Treat as unsupported type: Columns of any RAW data type are treated as an unsupported data type. If you select Stop on Any Bad Type as the Analysis Parameter, the data type mapping analysis stops when it encounters any RAW data type column. If you select any other option for the Analysis Parameter, the data type mapping analysis continues and RAW data type columns are ignored during table loading.

–   Map to VAR*: This maps columns of LOB data types from the Oracle database to VAR data types in the TimesTen database:

`RAW(n)` as `VARBINARY(n)` where *n* can have a maximum value of `4194304`.

`LONG` as `VARCHAR2(4194304)`

`LONG RAW` as `VARCHAR2(4194304)`

■   **Inline Column Size Limit**: The maximum size limit for inline columns. This is an integer value with valid values between `0` and `32768`.

■   **Padding Factor**: The percentage increase of the determined maximum value for precision and length as determined by analysis when Aggressive Mapping is selected for the `NUMBER` and `VARCHAR2` data type mapping. This is an integer value with valid values between `0` and `1000`. A padding factor of `50` indicates an increase of 50%.

Locate the Compression section.

2.   In the Compression section, set the following compression parameters:

■   **Type of Compression**: You can select the following types of compression:

–   **No Compression**: The source Oracle database table is not analyzed for compression. If you select this option, all of the compression parameters are disabled.

–   **Use Maximum Number of Distinct Values**: The source Oracle database table is analyzed for compression using the maximum dictionary pointer size for every column.

–   **Use Optimal Number of Distinct Values**: The source Oracle database table is analyzed for compression using a dictionary pointer size that is adjusted based on the actual number of distinct values in the column.

■   **Padding Factor**: The percentage increase of the determined maximum value for distinct values in a table as determined by analysis. This is an integer value

with valid values between `0` and `1000`. A padding factor of `50` indicates an increase of 50% for the `MAXVALUES` parameter of the `COMPRESS` statement. For more information on the `MAXVALUES` parameter, see "CREATE TABLE" in the *Oracle TimesTen In-Memory Database SQL Reference.*

■ **Minimum Compression Ratio**: The minimum compression factor that must be achieved for a column or table to be considered for compression. The value is the percentage ratio between the compressed and uncompressed size of the column or table. Smaller values represent higher compression. This is an integer value with valid values between `1` and `100`.

■ **Minium Number of Rows**: The minimum number of rows for table to be considered for compression. This is an integer value with valid values greater than `0`.

Locate the Analysis Parameters section.

3. Choose the values for the Analysis Parameters:

   ■ Select the action to take when an unsupported data type is encountered for a table column. There are two options:

      – **Stop on Any Bad Type**: The data type mapping analysis stops when it encounters a bad data type. If a bad data type is encountered, you can view the error message in analysis results.

      – **Skip Columns with Bad Types**: The data type mapping analysis skips columns with bad data types, but the analysis does not stop.

   Once you have set all the parameters of the Type Mapping & Compression tab, locate the Load Options tab.

4. Click the **Load Options** tab.

Figure 7–3   Click Load Options
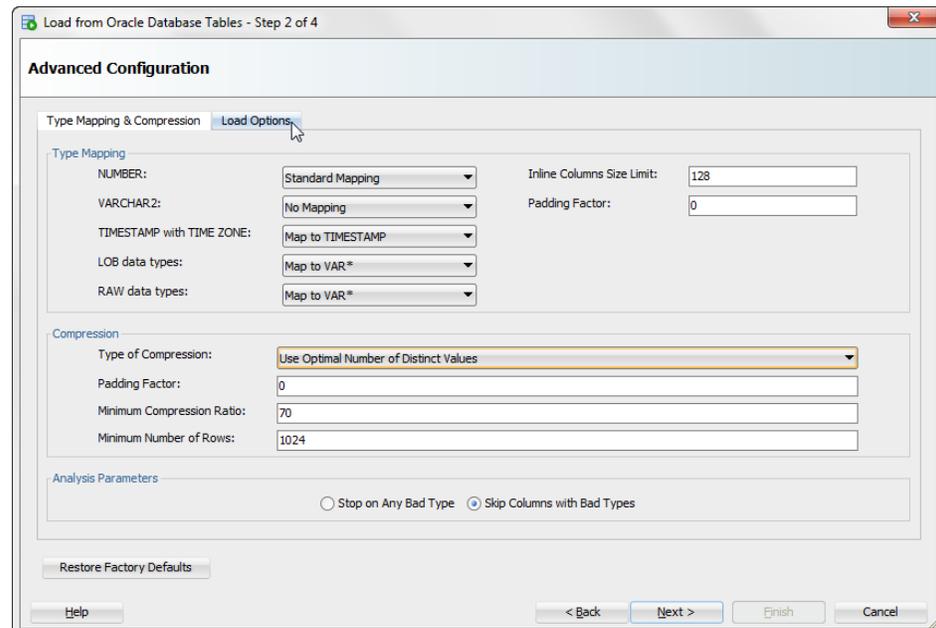
The Load Options tab displays. Locate the Complex Data Mapping section.

5. In the Load Options Configuration section of the Load Options tab, specify these options:

> **Note:** These options are only supported in TimesTen 18.1 or higher.

- **Inserter Threads**: Number of threads that insert data in the TimesTen table.
- **Fetcher Threads**: Number of threads that retrieve data from the Oracle database table.
- **Error Threshold**: Threshold for the maximum number of allowed errors.
- **Resume from SCN**: The load queries the Oracle database from this SCN. This value is optional and if it is not specified it will use the latest SCN.
- **Ignore Duplicates**: This option ignores any duplicate data. It only works if the TimesTen table has a uniqueness constraint on it. All uniqueness constraint violations are ignored.
- **Direct Load**: This option loads data directly into the table during commit points, whereas not using direct load loads data with standard `INSERT` SQL statements. The direct load has better performance and scalability for large amounts of data.

Locate the Other Parameters section in the Advanced tab.

6. In the Index Creation section, specify index creation policies:

- **Policy**: Specify one of these index creation policies:

   Copy all indexes from peer Oracle database table

   Only copy unique indexes from peer Oracle database table

   Do not copy indexes from peer Oracle database table

- Only create indexes when tables are loaded with no errors: This option determines if the indexes are created after tables are loaded without errors.

Locate the **Next>** button.

7. Click **Next>**.

Figure 7–4   Click Next>
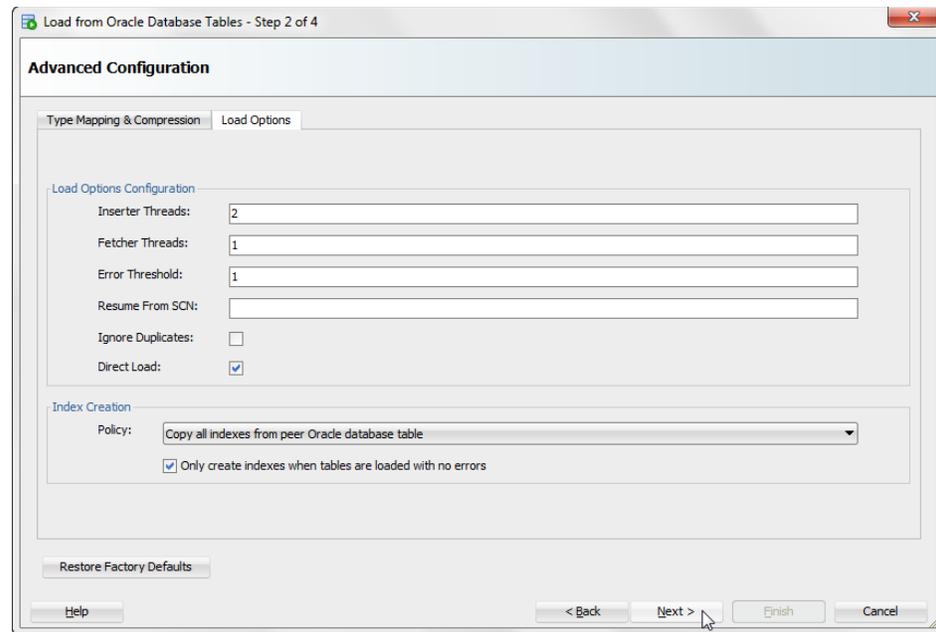
The Load from Oracle Database Tables: Load Selected Tables dialog displays. You can now follow the steps from "Simple" on page 7-7 as from this step forward, the steps are the same for the advanced and simple modes.

Simple

The simple mode enables you to load data from the Oracle database without selecting the mapping of Oracle database data types to TimesTen. However, if you are using advanced mode, you can follow these instructions after specifying the mapping of your data types and compression.

If you are using advanced mode, ensure that you have completed all the steps from "Advanced" on page 7-3.

If you are using simple mode, ensure that you have completed all the steps from "Loading data from Oracle database tables" on page 7-1.

In this dialog you need to select the tables from which you want to load data from the Oracle database. By default, the dialog shows the tables from your Oracle database schema. Optionally, you can perform these operations to help you select tables from other database schemas:

- Add an Oracle database schema
- Remove an Oracle database schema
- Apply a filter to Oracle database schema tables
- Load custom query results

If you do not want to select tables from other database schemas, you can move directly to the procedure for loading tables:

- Select tables to load

Add an Oracle database schema

To add tables from another Oracle database schema, follow these steps.

1. Click the + located above the Oracle database schema navigation tree.

Figure 7–5   Add an Oracle database schema
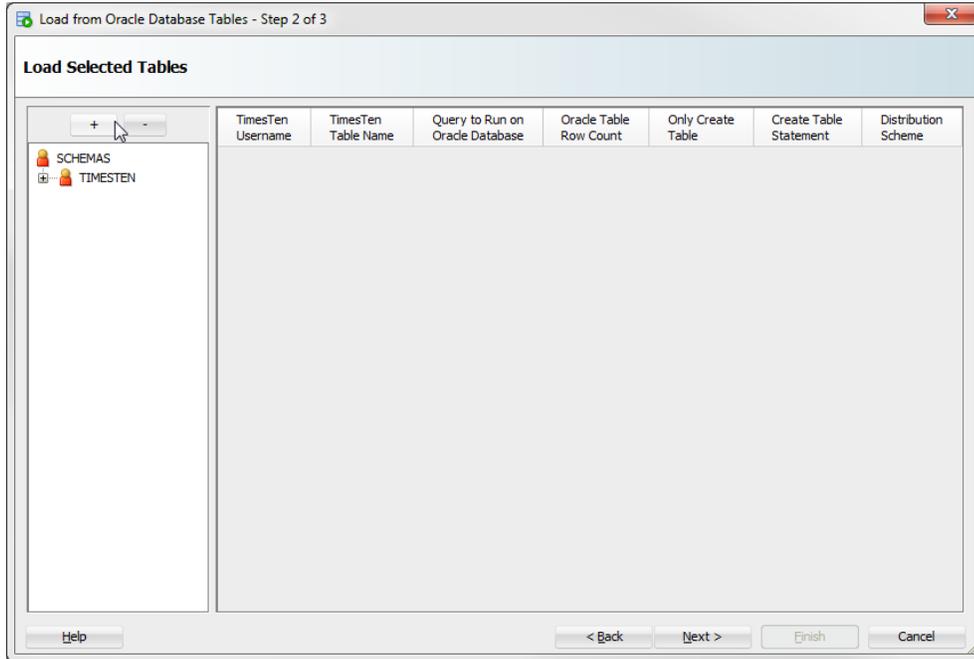
The **Choose Schema** dialog displays. Only Oracle database schemas that the connection user has access to display.

2. Select the Oracle database schema from the **Schema name** drop-down list.

Figure 7–6   Choose an Oracle database schema
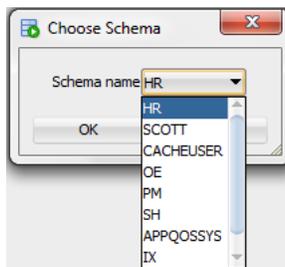
You are now ready to add the Oracle database schema.

3. Click **OK**.

You have successfully added the Oracle database schema. To load data from the tables of the Oracle database schema, see "Select tables to load" on page 7-12.

Remove an Oracle database schema

To remove an Oracle database schema, follow these steps. This only removes Oracle database schemas from being available for the Load from Oracle database feature. You cannot remove the Oracle database schema of the connection user.

1. Select the Oracle database schema that you want to remove in the schema navigation tree.

2. Click the - located above the Oracle database schema navigation tree.

Figure 7–7   Remove the Oracle database schema
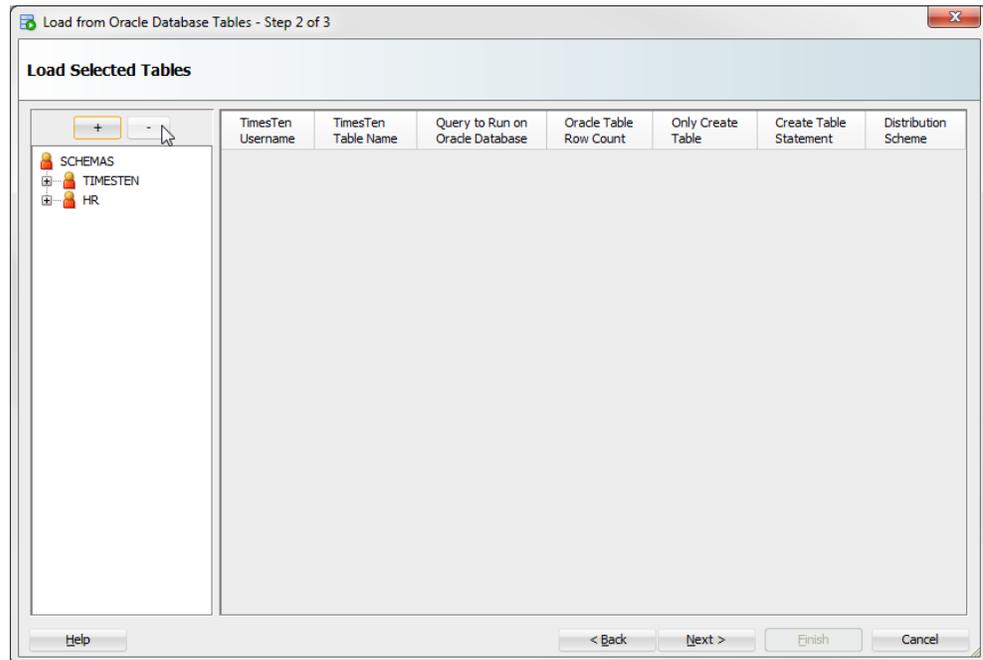
You have successfully removed the Oracle database schema. To load data from the tables of an Oracle database schema, see "Select tables to load" on page 7-12.

Apply a filter to Oracle database schema tables

To apply a filter to the Oracle database schema tables, follow these steps.

1. Right-click the name of the Oracle database schema that you want to apply a filter to and select **Apply Filter**.

Figure 7–8   Apply Filter
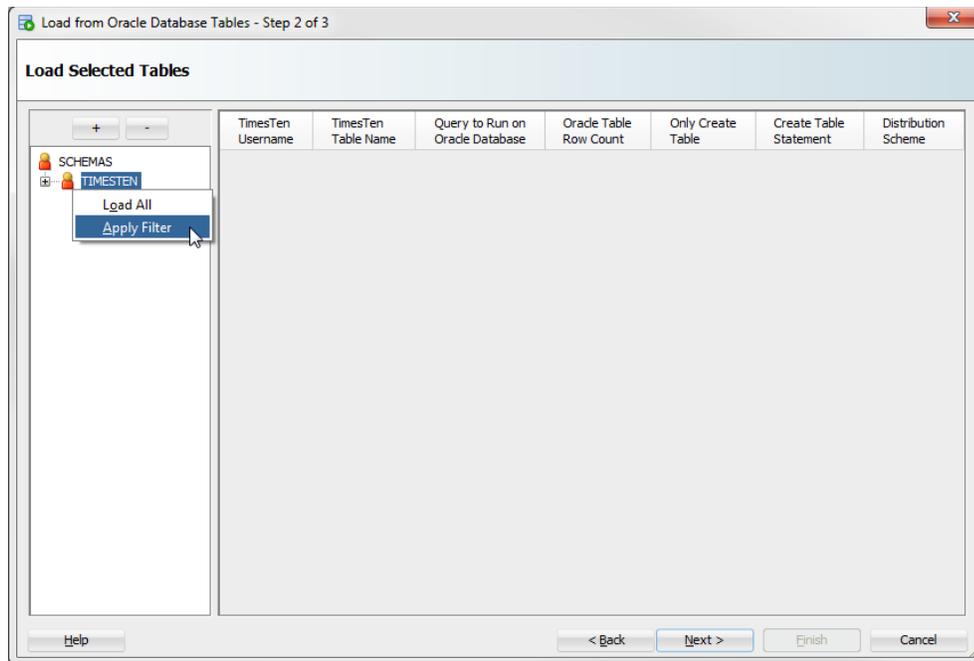
The **Filter** dialog displays.

2.   Select the filter condition from the **Table Name** drop-down list:

- **=**: Returns results that are equal to a specified pattern.

- **<>**: Returns results that are not equal to a specified pattern.

- **LIKE**: Returns results that match a specified pattern.

- **NOT LIKE**: Returns results that do not match a specified pattern.

Figure 7–9   Filter condition
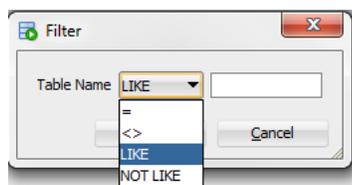
Locate the text field located next to the **Table Name** drop-down list.

3.   Specify a pattern for the filter condition in the text field located next to the **Table Name** drop-down list. You can use a wildcard for your filter:

- **_**: A substitute for exactly one character. For example `Sm_th` shows all tables that contain five characters, begin with `Sm`, and end with `th`.

- **%**: A substitute for zero or more characters. For example `EMP%` shows all tables that begin with the letters `EMP`.

4.   Click **OK**.

You have successfully applied a filter to the Oracle database schema tables. To load data from the tables of an Oracle database schema, see "Select tables to load" on page 7-12.

Load custom query results

To load custom query results for an Oracle database schema, follow these steps.

1. Right-click the **Schemas** option and select **Load custom query results**.

Figure 7–10   Load custom query results
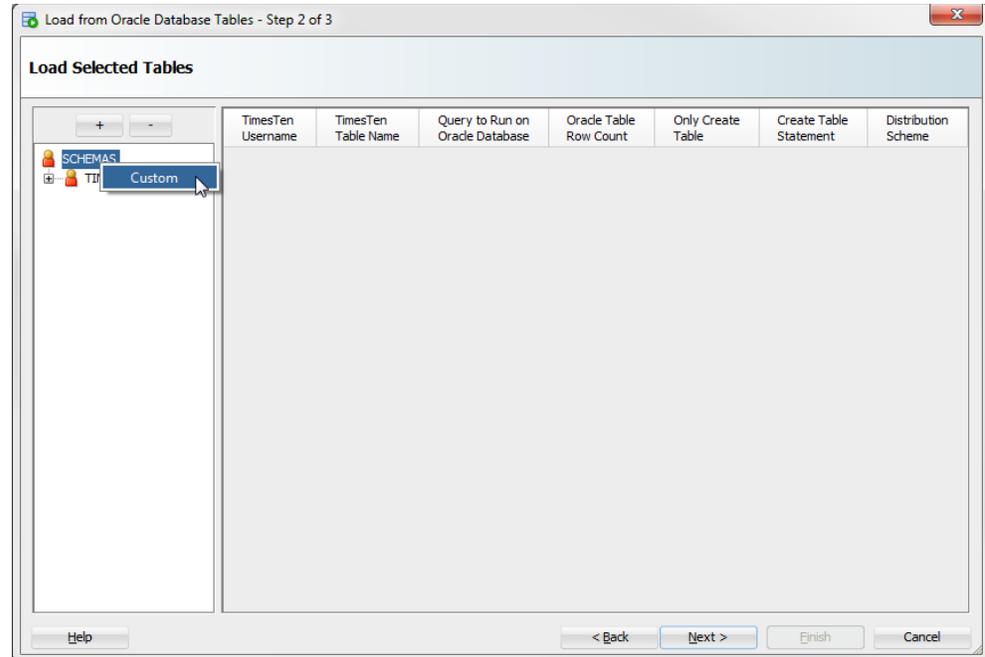
The **Custom Query** dialog displays. You are now required to enter the information for your custom Oracle query. The first property is the **Schema name**. You cannot edit this property and the default value is the connection user.

2. In the **TimesTen Username** text field, enter the owner of the TimesTen table. Ensure that the table owner you specify exists in TimesTen and that the TimesTen connection user has privileges to create a table or insert into an existing table owned by the specified table owner.

3. In the **TimesTen Table Name** text field, enter the name of the TimesTen table.

4. In the **Select Query** text field, enter the SQL query that you want to execute on the Oracle database to generate the desired result set. For more information on valid Oracle Database SQL queries, see *Oracle Database SQL Language Reference*.

5. In the **Parallel Thread Count** text field, enter the number of parallel threads that will be used to load the Oracle database table into the TimesTen database. TimesTen recommends a value of four.

6. Click **OK**.

Figure 7–11   Load the custom query
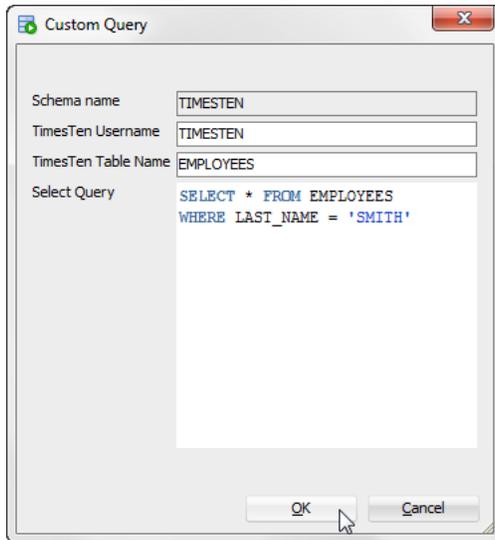
You have successfully loaded a custom Oracle database query. Your custom query is loaded into the right pane of the Load Tables from Oracle dialog. To load data from the tables of an Oracle database schema, see "Select tables to load" on page 7-12.

Select tables to load

To select the tables that you want to load from the Oracle database into the Timesten database.

1. Click the + located next to your Oracle database schema name to expand the Oracle database tables list.

Figure 7–12   Expand the Oracle database schema name
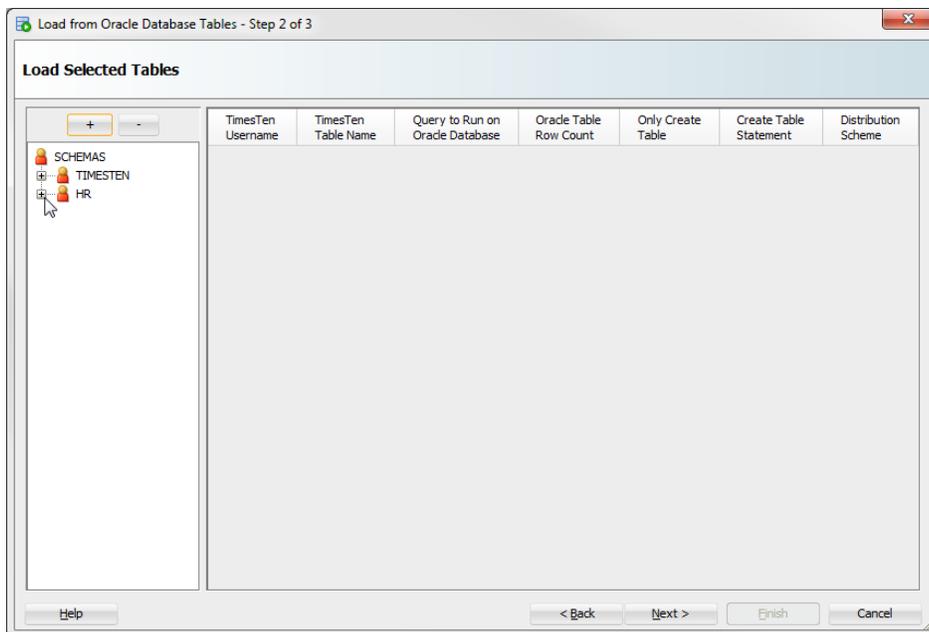
Your Oracle database schema expands showing a list of available Oracle database tables. Locate the Oracle database tables you want to load into the TimesTen database.

2. Right-click the name of the Oracle database table that you want to load into the TimesTen database and select **Load**. You can repeat this step, including for the same Oracle database table, to load multiple Oracle database tables.

   If you want to load all of the Oracle database tables, right-click the Oracle database schema name and select **Load All**. The TimesTen Username dialog displays. Enter the TimesTen user for which you want to create the tables and click **OK**.

Figure 7–13 Load an Oracle database table
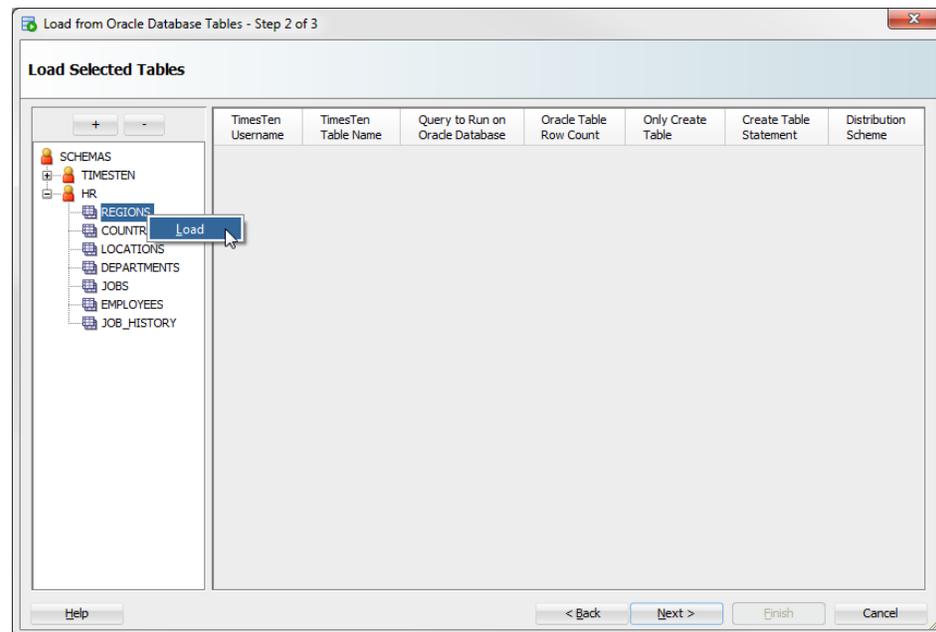
Your selected tables are added to the right pane and are displayed in a table format.

3. Review the selected Oracle database tables and edit any necessary fields. You can edit a field by double clicking it. A description of each column follows:

   ■ **TimesTen Username**

   The owner of the TimesTen table that you are creating. By default, this is the TimesTen connection user. You can edit this column. An error message is displayed if the specified table owner does not exist or the TimesTen connection user does not have privileges to access the specified table owner.

   ■ **TimesTen Table Name**

   The name of the TimesTen table that you are creating. By default, this is the same table name as the Oracle database table. You can edit this column.

   ■ **Query to Run on Oracle database**

   The SQL query executed on the Oracle database to generate the desired result set. By default, this is a `SELECT * FROM` *oracledb_tbl* query, where *oracledb_tbl* is the Oracle database table from which you are loading data. You can edit this column. For more information on valid Oracle Database SQL queries, see *Oracle Database SQL Language Reference*.

- **Oracle Table Row Count**

  The number of rows that TimesTen loads from the Oracle database table. This is the row count of the result set from the "Query to Run on the Oracle database" field. By default, this value is not counted.

  To populate the **Row Count** field for a specific Oracle database table, right-click on the **Row Count** field for your table, then select **Count**.

  To populate the **Row Count** fields for all the selected Oracle database tables, right-click on a **Row Count** field of any table, then select **Count All**. A confirmation dialog displays. Click **Yes** to continue with the row count of all of your previously selected Oracle database tables.

- **Only Create Table**

  This option enables you to choose if you want to only create a table and not load it with any data.

- **Create Table Statement**

  The SQL statement that TimesTen uses to create the table to load your Oracle database data. You cannot edit this column. Review the `CREATE TABLE` statement by double clicking on the **Create Table Statement** field.

  If a TimesTen table with the same name already exists in the TimesTen database, the data from the Oracle database table is appended to the existing TimesTen table. TimesTen does not load table constraints from the Oracle database table. An error dialog displays if a TimesTen table with the same name but different table structure already exists in the TimesTen database.

- **Distribution Scheme**

  This column is only displayed if you are using a TimesTen Scaleout database.

  Click the **View/Edit** button in this column to select a distribution scheme for this table.

  From the **Distribution method** drop-down list, select a distribution method for your table:

  Distribute by Hash - Distribution scheme that distributes data based on the hash of the primary key or a set of user-specified columns. For more information, see "Hash" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

  If you select **Distribute by Hash**, select columns to be part of your distribution column:

  To select columns to be part of your distribution column, double click the column name in the **Available Columns** list.

  To remove columns from your distribution column, double click the column name in the **Selected Columns** list.

  Distribute by Reference - Distribution scheme that distributes the data of a child table based on the location of the corresponding parent row of a foreign key constraint. Your table needs at least one foreign key to be able to use this distribution scheme. For more information, see "Reference" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

  If you select Distribute by Reference, from the **Foreign Key** drop-down list, select the foreign key to distribute the data of your table.

Duplicate - Distribution scheme that distributes full identical copies of the table's data to all the elements of a database. For more information, see "Duplicate" in the *Oracle TimesTen In-Memory Database Scaleout User's Guide*.

If you select **Duplicate**, you do not need to specify any additional fields in the Distribution tab of the Create Table dialog.

Once you have edited all of the fields, locate the **Next>** button at the bottom of the dialog.

4. Click **Next>**.

Figure 7–14   Click Next>
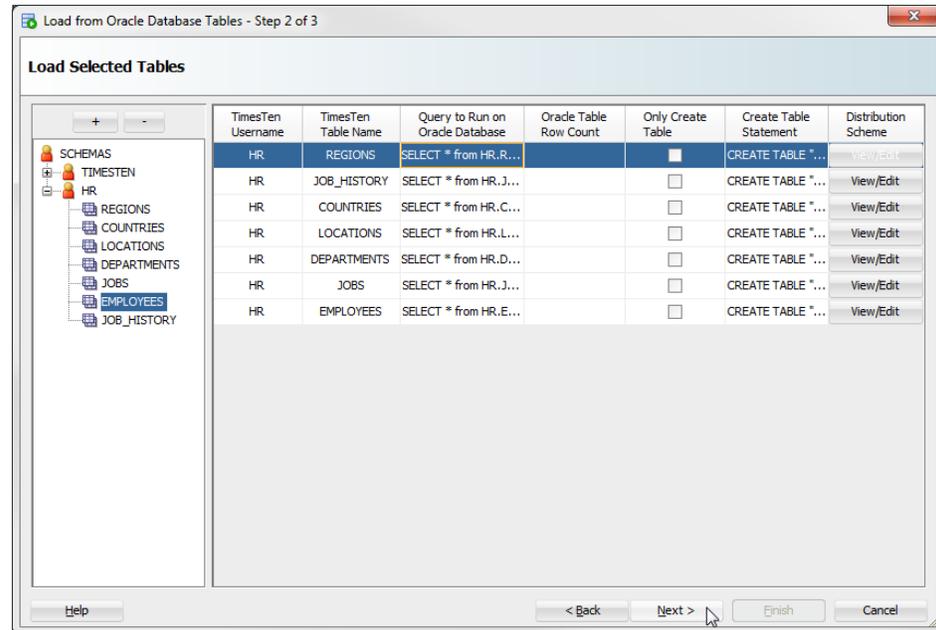
A Load From Oracle Database Tables progress dialog displays. Once TimesTen finishes loading your tables, the progress dialog closes and the Load Summary results displays. If there are any errors, troubleshoot the errors. Locate the Finish button.

5. Click **Finish**.

If your load operation had any errors, SQL Developer asks if you want to save the load profile. This load profile for resuming the load operation once you have fixed any errors that might have come up. See for more information on how to resume the load operation.

The Load Summary dialog closes.

You have successfully loaded data from an Oracle database into a TimesTen database.

Loading data from an Oracle database table into an existing TimesTen table

To load data from an Oracle database table into an existing TimesTen table, follow these steps. Ensure that you are on the main SQL Developer page and that your TimesTen connection is expanded.

1. Click the + located next to **Tables** to expand the TimesTen tables list.

   The TimesTen tables list expands.

2. Right-click the name of the table and select the **Table** option, then select the **Load from Oracle Database Table** option.

Figure 7–15   Load data from an Oracle database table
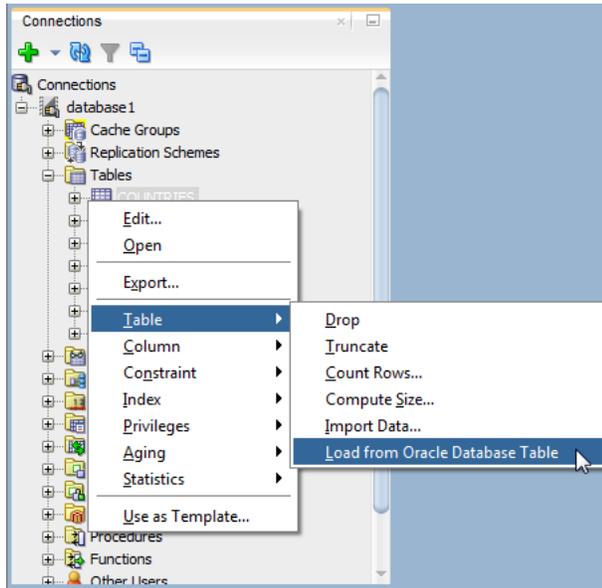
The **Load from Oracle Database Tables** dialog displays. The Schema name, TimesTen Username, and TimesTen Table Name fields are auto-filled and you cannot edit these fields. Prepare to enter the SQL query and the Parallel Thread Count.

3. In the **Select Query** field, enter the SQL query to execute on the Oracle database. The result of the SQL query is loaded into your TimesTen table. For more information on valid Oracle Database SQL queries, see *Oracle Database SQL Language Reference*.

4. In the Load Options section, specify these options:

   ___

   Note: These options are only supported in TimesTen 18.1 or higher.

   ___

   - **Inserter Threads**: Number of threads that insert data in the TimesTen table.

   - **Fetcher Threads**: Number of threads that retrieve data from the Oracle database table.

   - **Error Threshold**: Threshold for the maximum number of allowed errors.

   - **Resume from SCN**: The load queries the Oracle database from this SCN. This value is optional and if it is not specified it will use the latest SCN.

   - **Ignore Duplicates**: This option ignores any duplicate data. It only works if the TimesTen table has a uniqueness constraint on it. All uniqueness constraint violations are ignored.

   - **Direct Load**: This option loads data directly into the table during commit points, whereas not using direct load loads data with standard INSERT SQL

statements. The direct load has better performance and scalability for large amounts of data.

Locate the Load button.

5. Click **Load**.

*Figure 7–16   Load Tables from the Oracle database*
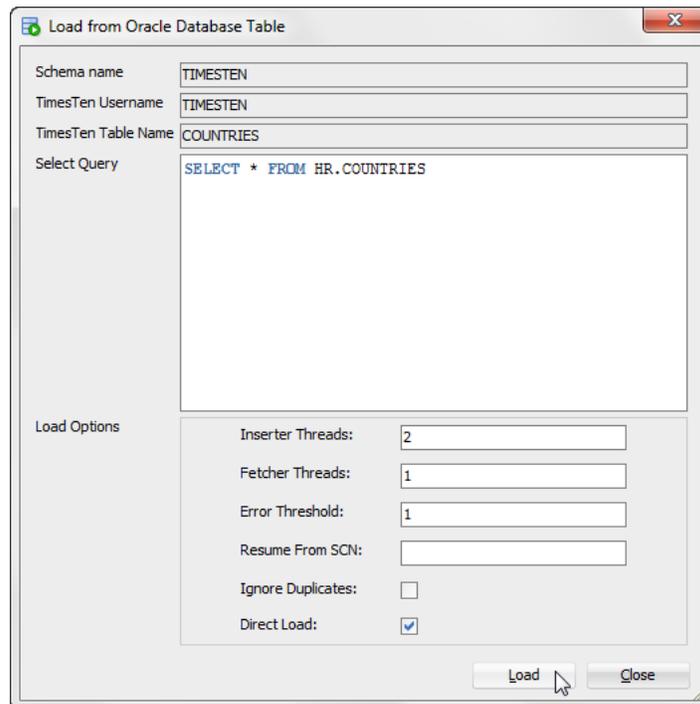
A progress dialog displays. Once TimesTen loads the data from the Oracle database table, the progress dialog closes. If there are any errors, troubleshoot the errors. If you execute the query on a small Oracle database table, the progress dialog may close almost immediately.

Resume loading data from Oracle database tables

In some cases when you attempt to load data from Oracle database tables, you may run into some errors that prevent all data from being loaded from the Oracle database tables. If you saved the load profile after your load operation failed, you can resume the load operation using the load profile.

To resume loading data from an Oracle database table, ensure that you saved the load profile, that you are on the main SQL Developer page, and that your TimesTen connection is expanded.

1. Right-click the **Table** option and select the **Resume Loads from Oracle Database Tables** option.

Figure 7–17   Select Resume Loads from Oracle Database Tables
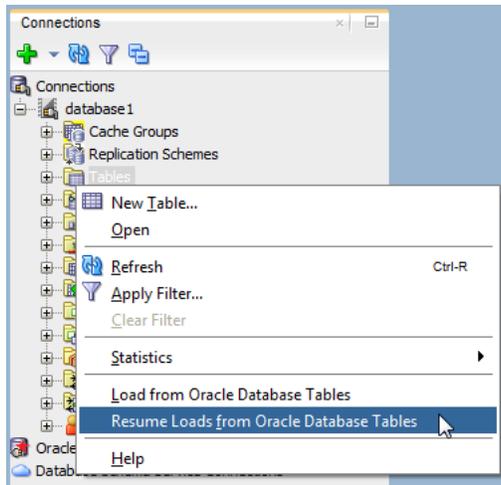
The Resume Loads from Oracle Database Tables dialog displays. Locate the Browse button.

2. Click **Browse**.

    The Open dialog displays. Locate your load profile file.

3. Once you have located your load profile file, click **Open**.

    The Open dialog closes.

    > **Note:**   SQL Developer displays a warning message if you are using a different database connection than the one that was used when the load profile was saved.

4. Review the information that is loaded in the Resume Loads and Load Options tabs.

    The Resume Loads tab displays information related to which tables can be loaded from the Oracle database tables into TimesTen tables.

    The Load Options tab displays information related to the load options that will be used to load the Oracle database tables into TimesTen tables. These options cannot be edited.

5. Once you have reviewed the load information, click **Resume Loads**.

Figure 7–18 Click Resume Loads
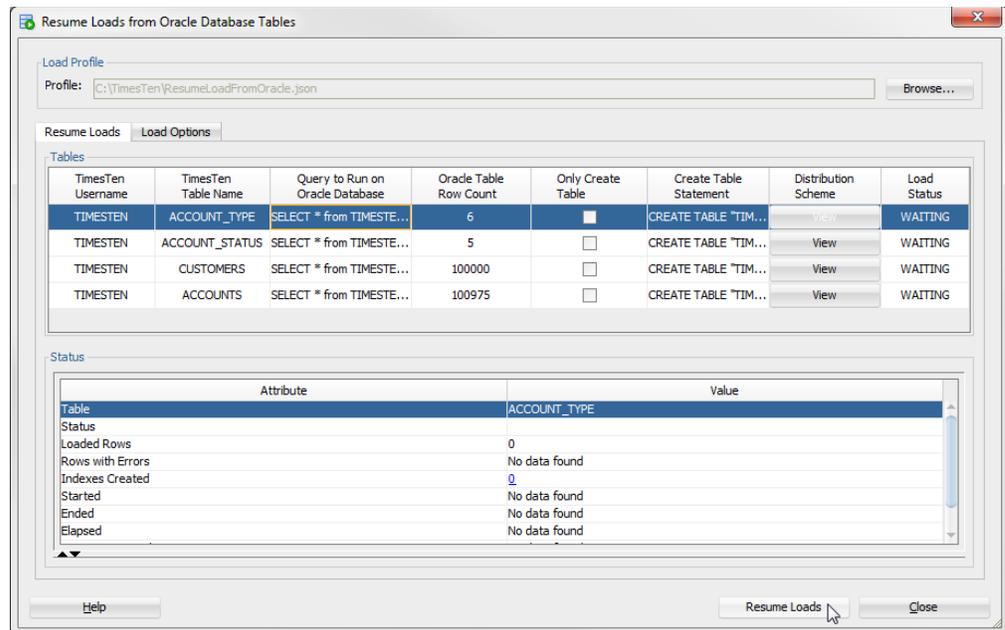
A progress dialog displays. Once all tables have been loaded, the progress dialog closes. If SQL Developer displays an error message, troubleshoot the error.

Locate the Close button.

6. Click **Close**.

The Resume Loads from Oracle Database Tables dialog closes.

You have successfully resumed loading data from Oracle Database tables.

8

Prerequisites to Creating and Accessing a Cache Environment

After you create the TimesTen database, and before you cache Oracle database data in the TimesTen database, you must perform the tasks:

> **Note:** TimesTen cache is not supported with TimesTen Scaleout databases.

- Configuring the Oracle and TimesTen databases
- Set the cache administration user name and password
- Start and stop the cache agent

Configuring the Oracle and TimesTen databases

Before you cache Oracle database data in a TimesTen database, you must perform certain tasks either outside of SQL Developer using command-line interfaces such as SQL*Plus and the `ttIsql` utility, or within SQL Developer using the **SQL Worksheet**.

The tasks include:

- Creating the timesten user, a schema user and the cache administration user in the Oracle database.
- Granting privileges required to perform cache operations to the Oracle database users.
- Creating the cache manager user and the cache table users in the TimesTen database.
- Granting cache privileges to the TimesTen users.

For information about creating Oracle database users and TimesTen users, see "Create the Oracle database users" and "Create the TimesTen users" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

For information on configuring Oracle and TimesTen databases, see "Configuring the Oracle database to cache data in TimesTen Classic" and "Configuring a TimesTen database to cache Oracle Database data" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

Set the cache administration user name and password

You must create a cache administration user in the Oracle database. This user enforces predefined behaviors of cache groups.

After you create the cache administration user in the Oracle database, you must set the name and password for the user in the TimesTen database.

To set the cache administration user name and password, right-click the node of the connection name for the TimesTen database and select **Set Cache Administration User**. See Figure 3–7, "Viewing a TimesTen connection".

The **Setting the cache administration user** dialog appears. For the **Prompts** tab, in the **Cache administration user ID** field, enter the cache administration user name. In the **Cache administration user password** field, enter the Oracle database password for the cache administration user.

Click **Apply**.

Figure 8–1 Setting the cache administration user dialog
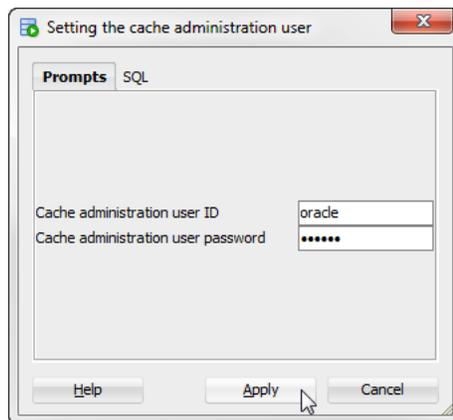
You must set the cache administration user name and password for each TimesTen database that caches an Oracle database. If you drop and re-create the TimesTen database or drop and re-create the cache administration user in the Oracle database, then the cache administration user name and password must be reset in the TimesTen database.

For more information about the cache administration user, see "Set the cache administration user name and password" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

Start and stop the cache agent

After you set the cache administration user name and password, you must start the TimesTen cache agent process. The cache agent processes cache operations such as loading a cache group and automatic refresh.

To start the cache agent, right-click the node of the connection name for the TimesTen database and select **Start/Stop Cache Agent**. See Figure 3–7, "Viewing a TimesTen connection".

In the **Starting/stopping the cache agent** dialog box, select **Start agent** within the **Prompts** tab. Then click **Apply** to start the cache agent.

Figure 8–2   Starting/stopping the cache agent
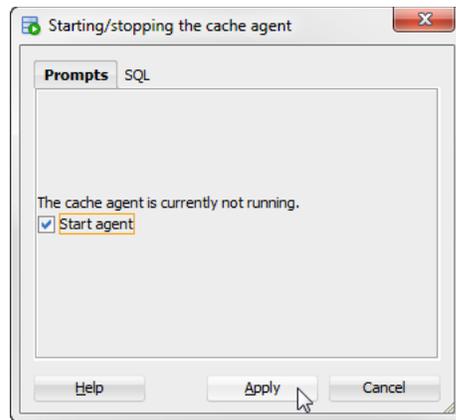
To stop a running cache agent process, right-click the node of the connection name for the TimesTen database and select **Start/Stop Cache Agent** in an identical fashion as was done when starting the cache agent.

In the **Starting/stopping the cache agent** dialog box, select **Stop agent** within the **Prompts** tab. Then click **Apply** to stop the cache agent. See Figure 8–2, "Starting/stopping the cache agent".

For more information about the cache agent process, see "Managing the cache agent" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

9

Creating Cache Groups

A cache group defines the set of Oracle database data to cache in a TimesTen database. When you create a cache group, cache tables are created in the TimesTen database that correspond to the Oracle database tables being cached.

> **Note:** TimesTen cache is not supported with TimesTen Scaleout databases.

This chapter includes topics:

- Creating a cache group including:
    - Defining a cache group and cache table attributes
    - Defining the root table
    - Defining the child tables
- Dropping a cache group

Creating a cache group

To create a cache group, right-click the **Cache Groups** node under the node of the connection name for the TimesTen database and select **New Cache Group**.

Figure 9–1 Creating a new cache group
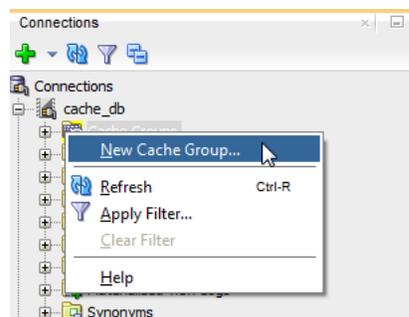
In the **Create cache group** dialog, specify a name for the cache group in the **Name** field within the **Cache Group Attributes** tab. For **Type**, choose the desired cache group type (read-only, asynchronous writethrough, synchronous writethrough or user managed).

Cache groups are categorized as either explicitly loaded or dynamic. If you are creating a dynamic cache group, then choose **Dynamic** in the **Type** section.

In an explicitly loaded cache group, data is loaded manually into its cache tables from the corresponding Oracle database tables using a load or refresh operation, or automatically using an automatic refresh operation. In a dynamic cache group, data is loaded into its cache tables on demand from the corresponding Oracle database tables using a dynamic load operation, or manually using a load operation.

A read-only, asynchronous writethrough (AWT) or synchronous writethrough (SWT) cache group can be defined as a dynamic cache group. A user managed cache group can be defined as a dynamic cache group if it does not contain all of the following characteristics:

- Automatic refresh is defined.

- At least one of its cache tables are updatable.

- Committed updates on one or more of its updatable cache tables are propagated to the corresponding Oracle database tables.

For more information about explicitly loaded and dynamic cache groups, see "Loading data into a cache group: Explicitly loaded and dynamic cache groups" and "Dynamic cache groups" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

Figure 9–2   Defining cache group type
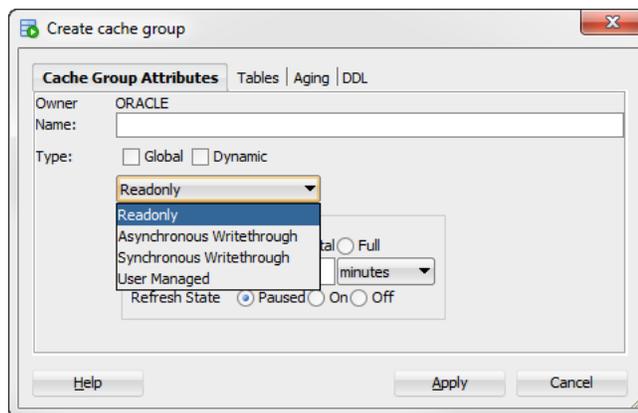
Defining a cache group and cache table attributes

Within the **Cache Group Attributes** tab of the **Create cache group** dialog, if you are creating a user managed cache group and you select **Autorefresh** within the **Table Attributes** section to enable automatic refresh when either **All tables readonly** or **Propagate all tables** is selected, or you are creating a read-only cache group, then for the following automatic refresh characteristics:

- Set the automatic refresh mode in the **Mode** field by choosing either **Full** or **Incremental**. The default automatic refresh mode is Incremental.

- Set the automatic refresh interval in the **Refresh Interval** field by specifying a numeric value followed by a unit of time in minutes, seconds or milliseconds. The default automatic refresh interval is 5 minutes.

- Set the automatic refresh state in the **Refresh State** field by choosing either **Paused**, **On** or **Off**. The default automatic refresh state is Paused.

Figure 9–3   Autorefresh attributes
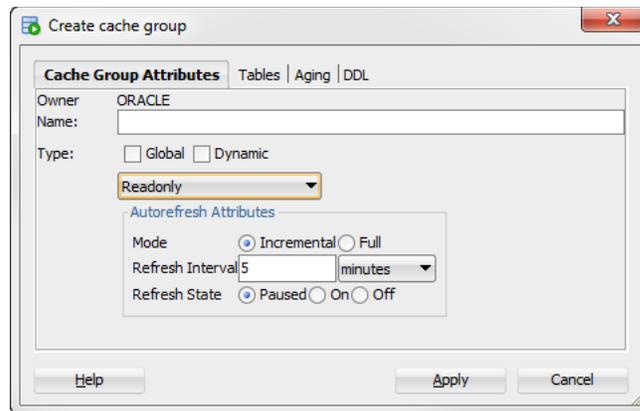
For more information about automatic refresh of cache groups, see "AUTOREFRESH cache group attribute" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

For more information about read-only cache groups, see "Read-only cache group" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

If you are creating a user managed cache group, then from the **Table Attributes** section in the **Cache Group Attributes** tab within the **Create cache group** dialog, choose one:

- **All tables readonly**: All cache tables are read-only.

- **Propagate all tables**: All cache tables are updatable and committed updates on the cache tables are propagated to the corresponding Oracle database tables.

- **Varies from table to table**: Some cache tables are read-only and other cache tables are updatable where committed updates on the cache tables may or may not be propagated to the corresponding Oracle database tables.

Figure 9–4   User managed cache group attributes
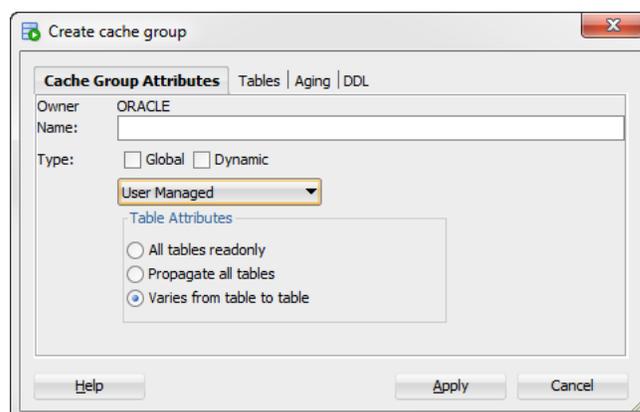
For more information about user managed cache groups, see "User-managed cache group" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

Defining the root table

Click the **Tables** tab in the **Create cache group** dialog. Right-click in the **Tables** panel and select **Add root table**.

Figure 9–5   Adding the root table
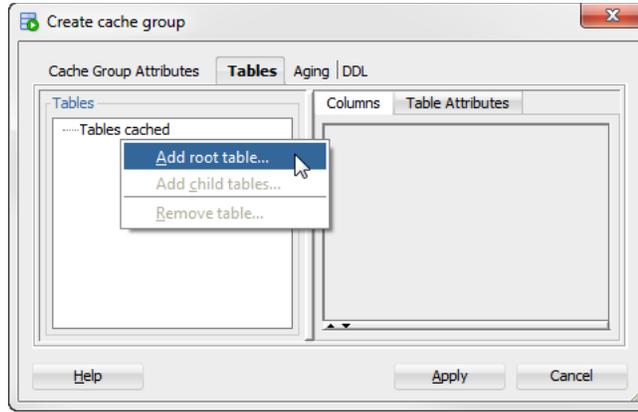
The **Select the root table** dialog appears. The list of Oracle database tables and synonyms that you can choose to designate as the root table for your cache group is displayed. Oracle database tables that have primary keys and Oracle database tables that have unique non-null indexes are available for selection.

Select the table and click **OK**.

Figure 9–6   List of tables and synonyms for root table
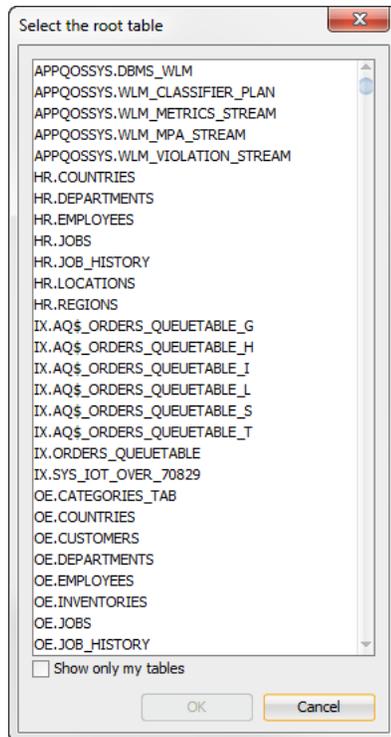
To show only Oracle database tables owned by the current connection user, choose **Show only my tables**.

For more information on caching Oracle database synonyms, see "Caching Oracle Database synonyms" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

Selecting the columns to cache

In the **Columns** tab within the **Tables** tab of the **Create cache group** dialog, select the check boxes under the **Inline** field to specify the INLINE attribute for the column. You can specify the INLINE attribute for columns of type VARCHAR2, NVARCHAR2, and VARBINARY.

Select the check boxes under the **Allow null values** field for each cached column to accept null values. By default, the nullability of a cache table's column is identical to the nullability of the corresponding Oracle database table's column.

Select the check boxes under the **Cached** field for each column to cache. By default, all columns of the table are selected and are cached. The Oracle database table's unique identifying columns (primary key, unique index or unique constraint) must be included in the TimesTen cache table and constitutes the cache table's primary key.

To select all columns or deselect all columns, except for the Oracle database table's unique identifying columns, to accept null values or to cache, select or deselect the check boxes that appear to the left of the **Allow null values** and **Cached** fields, respectively.

For more information about the characteristics of cache tables and cache groups, see "Cache groups and cache tables" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

Some Oracle database data types have multiple compatible TimesTen data types. For columns in the Oracle database table whose data type can map multiple TimesTen data types, a menu appears under the **Data type in TimesTen** field for those columns. Select the desired data type for the column in the cache table. The default compatible TimesTen data type for the Oracle database data type appears in italics.

For more information about compatibility between Oracle database and TimesTen data types, see "Mappings between Oracle Database and TimesTen data types" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

For updatable cache tables, you can specify a default column value under the **Default Value** field for each cached column to define a default value. By default, the default value of a cache table's column is identical to the default value of the corresponding Oracle database table's column.

Specifying the table attributes

By default, TimesTen creates a range index on the cache table's primary key columns. To create a hash index instead of a range index on the primary key columns, in the **Primary Key** section of the **Table Attributes** tab within the **Tables** tab of **Create cache group**, choose **Use hash**. You must size the hash index by specifying either the maximum number of rows the underlying cache table is anticipated to have or the number of pages to allocate to the index in the **Hash sizing policy** menu and the adjacent field after the colon (**:**) which requires a numeric value. The default selection is **Current table row count**.

Figure 9–7   Primary key attributes
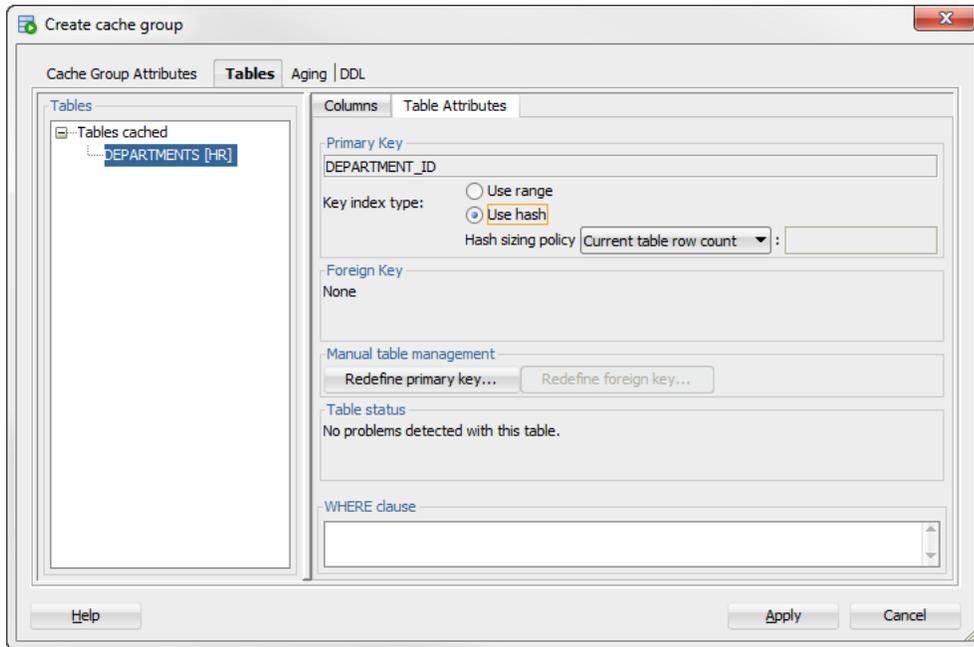
For more information about the index that is automatically created on a cache table's primary key columns, see "CREATE CACHE GROUP" in the *Oracle TimesTen In-Memory Database SQL Reference*.

For more information about properly sizing a hash index, see "CREATE TABLE" in the *Oracle TimesTen In-Memory Database SQL Reference*.

By default, the primary key columns of a cache table are identical to the primary key columns of the corresponding Oracle database table. You can define a different column or set of columns as the primary key for the cache table.

In the **Manual table management** section of the **Table Attributes** tab within the **Tables** tab of the **Create cache group** dialog, click **Redefine primary key**.

Figure 9–8   Redefining a primary key
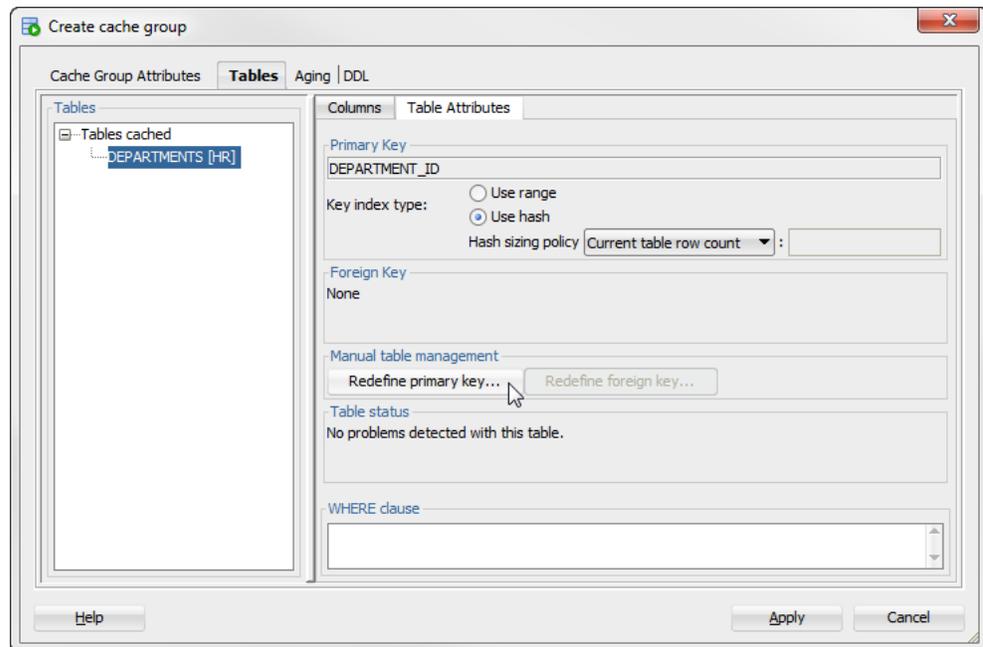
In the **Define Primary Key(s)** dialog, the non-nullable unique columns of the corresponding Oracle database table is shown in the **Non-null unique columns** list. Select the columns for the primary key of the cache table and then choose **>>** to move those columns into the **Primary key(s)** list. The order of the columns in a composite primary key is significant if a range index is to be created on the columns. Click **OK** to define the alternate primary key for the cache table.

Figure 9–9   Defining a new primary key
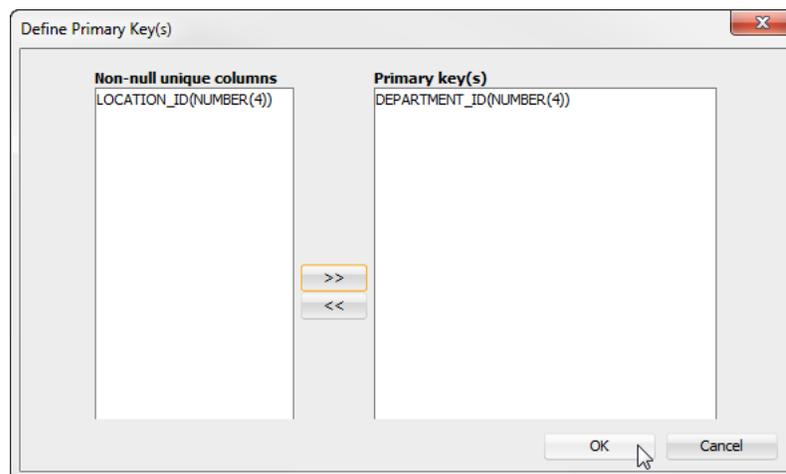
If the primary key of the cache table consists of a different set of columns than the primary key of the corresponding Oracle database table, then the tables may become unsynchronized when committed updates are made on the cache table or the corresponding Oracle database table. Unique constraint violations can occur when:

■   Committed updates on the cache table are propagated to the corresponding Oracle database table.

■ Committed updates on the Oracle database table are refreshed to the corresponding cache table.

You can specify an optional WHERE clause for the cache table in the **WHERE clause** field of the **Table Attributes** tab within the **Tables** tab of **Create cache group**. A WHERE clause is only allowed for table definitions in a read-only cache group or a user managed cache group that has automatic refresh defined.

Figure 9–10   *Specifying a where clause*
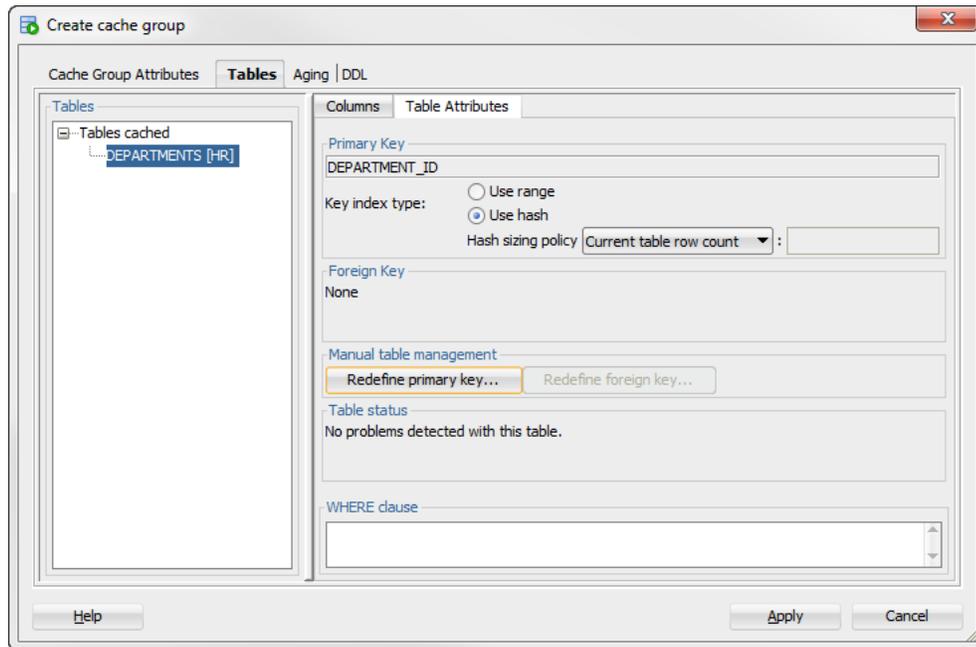
For more information about specifying a WHERE clause in the table definition of a cache group, see "Using a WHERE clause" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

If the cache table is in a user managed cache group that has selected the **Varies from table to table** in the **Table attributes** section within the **Cache Group Attributes** tab of the **Create cache group** dialog, then you must specify the table attribute for the cache table.

In the **Propagation Type** section of the **Table Attributes** tab within the **Tables** tab of **Create cache group**, choose:

■ **Readonly** to define the cache table as read-only

■ **Propagate** to define the cache table as updatable with updates on the table to be propagated to the corresponding Oracle database table.

■ **Neither** to define the cache table as updatable with updates on the table not propagated to the corresponding Oracle database table.

By default, cache tables in a user managed cache group are updatable with updates on the table not propagated to the corresponding Oracle database table.

Specifying an aging policy

You can specify an aging policy for the cache table in the **Aging** tab of the **Create cache group** dialog. An aging policy is explicitly defined on the root table of a cache group and it implicitly applies to all the child tables.

Choose:

- **Usage-based** to define a least recently used (LRU) aging policy
- **Time-based** to define a time-based aging policy
- **No aging** to not define an aging policy

An LRU aging policy can be defined for tables in an explicitly loaded AWT, SWT or user managed cache group that does not have automatic refresh defined. The default aging state is set to off for explicitly loaded cache groups. For dynamic cache groups that do not have automatic refresh defined, the default aging state is set to on and the type of aging is set to usage-based.

The TimesTen database memory usage thresholds that determine when data starts and stops being deleted from tables that are subject to LRU aging are shown when the **Usage-based** is selected. The default usage threshold that determines when data starts being deleted from the tables is 90%. The default usage threshold that determines when data stops being deleted from the tables is 80%. The default LRU aging cycle is 1 minute.

For information about how to change the memory usage thresholds and LRU aging cycle, see "Changing the memory usage thresholds and LRU aging cycle" on page 4-23.

Figure 9–11   Specifying usage-based aging
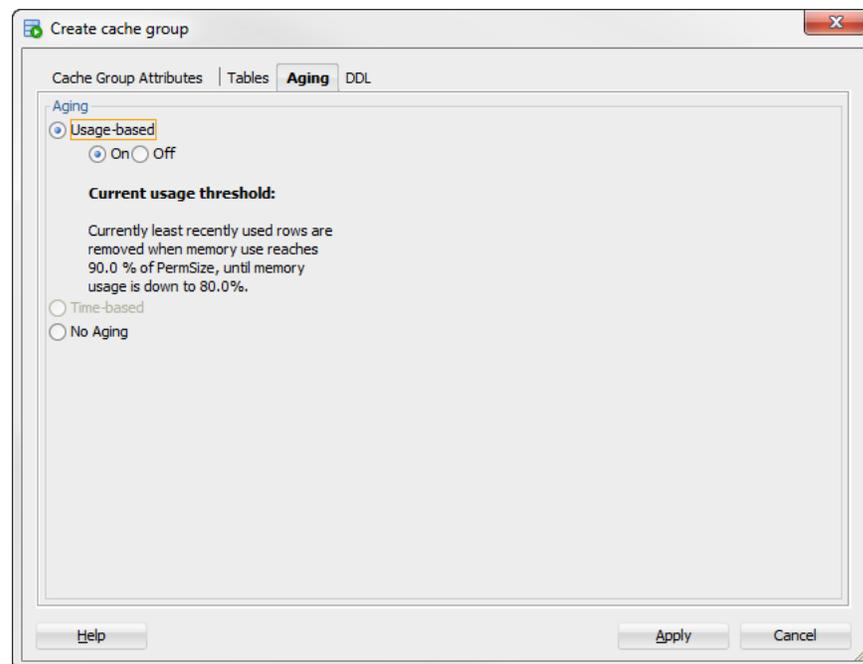
For more information about defining an LRU aging policy on cache tables, see "LRU aging" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

A time-based aging policy can be defined on a cache table that contains a non-nullable DATE or TIMESTAMP column. Columns in the table of these data types appear in the **Column** within the **Aging** tab of the **Create cache group** dialog when **Time-based** is selected. Select the column to store the timestamp value indicating when each row was inserted or most recently updated.

In the **Life-time** field, indicate the length of time in which data that has not been updated is to be kept in the table by specifying a numeric value followed by a unit of time in minutes, hours or days from the drop-down menu list.

In the **Cycle** field, indicate the frequency in which data is to be aged out of the table by specifying a numeric value followed by a unit of time in minutes, hours or days in the drop-down menu list. The default time-based aging cycle is 5 minutes.

Specify an aging state by choosing **on** or **off**.

Figure 9–12   Specifying time-based aging
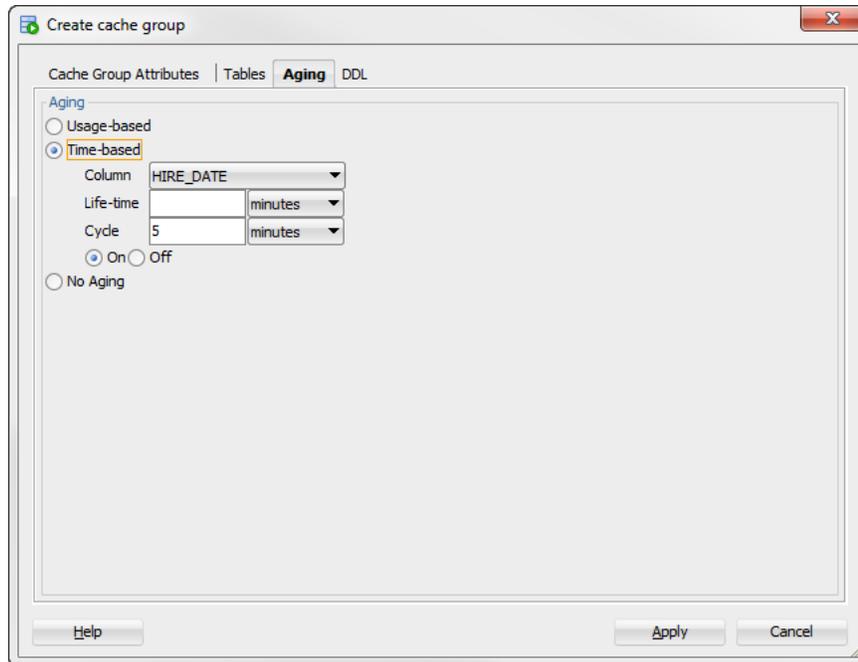
For more information about defining a time-based aging policy on cache tables, see "Time-based aging" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

Verifying the root table definition

After you have defined the tables and columns to cache, the type of index to create on the primary key columns, the table attributes including an optional WHERE clause, and an optional aging policy for the root table, you can view the CREATE CACHE GROUP statement used to create the cache group and its cache table in the **DDL** tab of the **Create cache group** dialog.

Click **Apply** to create the cache group and its cache table if you are not going to add any child tables to the cache group.

Figure 9–13   Verifying the cache group definition
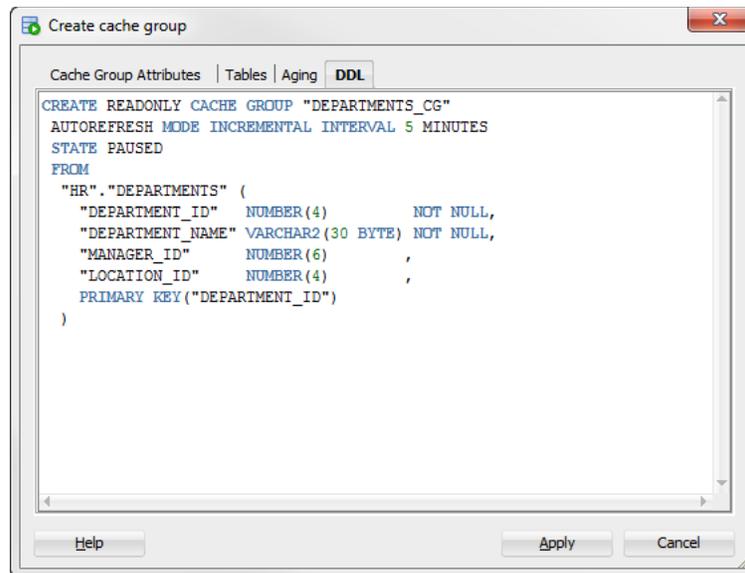
For information about the syntax and semantics of the CREATE CACHE GROUP statement, see "CREATE CACHE GROUP" in the *Oracle TimesTen In-Memory Database SQL Reference*.

Defining the child tables

Within the **Tables** tab of the **Create cache group** dialog, right-click in the **Tables** panel and select **Add child tables**.

The **Select one or more child tables** dialog appears. This dialog displays the list of Oracle database tables and Oracle database synonyms that you can choose to designate as the child table(s) for your cache group. Oracle database tables that have primary keys and Oracle database tables that have unique non-null indexes are available for selection.

Select the tables to designate as the child tables for your cache group and click **OK**.

For more information on caching Oracle database synonyms, see "Caching Oracle Database synonyms" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

Figure 9–14   List of table and synonyms for child tables
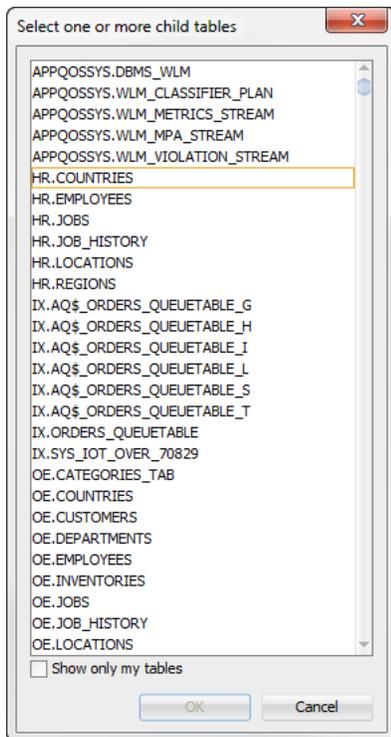
To select multiple child tables, press and hold the CTRL key, and click the desired tables. The child tables appear under their appropriate parent table. You can also drag and drop a particular child table so that it appears under the desired parent table.

To show only Oracle database tables owned by the current connection user, choose **Show only my tables**.

Selecting the columns to cache

Click a child table and in the **Columns** tab within the **Tables** tab of the **Create cache group** dialog, select the check boxes under the **Allow null values** field for each cached column to accept null values. By default, the nullability of a cache table's column is identical to the nullability of the corresponding Oracle database table's column.

Select the check boxes under the **Cached** field for each column to cache. By default, all columns of the table are selected and are cached. The Oracle database table's unique identifying columns must be included in the TimesTen cache table and constitute the cache table's primary key. The Oracle database table's foreign key columns that reference its parent table must also be included in the cache table.

To select all columns or deselect all columns, except for the Oracle database table's unique identifying columns and the foreign key columns that reference its parent table, to accept null values or to cache, select or deselect the check boxes that appear to the left of the **Allow null values** and **Cache** fields.

For columns in the Oracle database table whose data type can map to multiple TimesTen data types, a drop-down select list appears under the **Data type in TimesTen** field for those columns. Select the desired data type for the column in the child cache table.

For updatable cache tables, you can specify a default column value under the **Default Value** field for each cached column in the child cache table that defines a default value.

Specifying the table attributes

In the **Table Attributes** tab within the **Tables** tab of the **Create cache group** dialog, select whether to create a range index or a hash index on the child cache table's primary key columns in a similar fashion as was specified for the root table. You can define a different column or set of columns as the primary key for the cache table than the columns of the corresponding Oracle database table in a similar fashion as was specified for the root table. You can also specify an optional WHERE clause for the cache table in a similar fashion as was specified for the root table. A WHERE clause is only allowed for table definitions in a read-only cache group or a user managed cache group that has automatic refresh defined.

Figure 9–15   Specifying attributes for child table
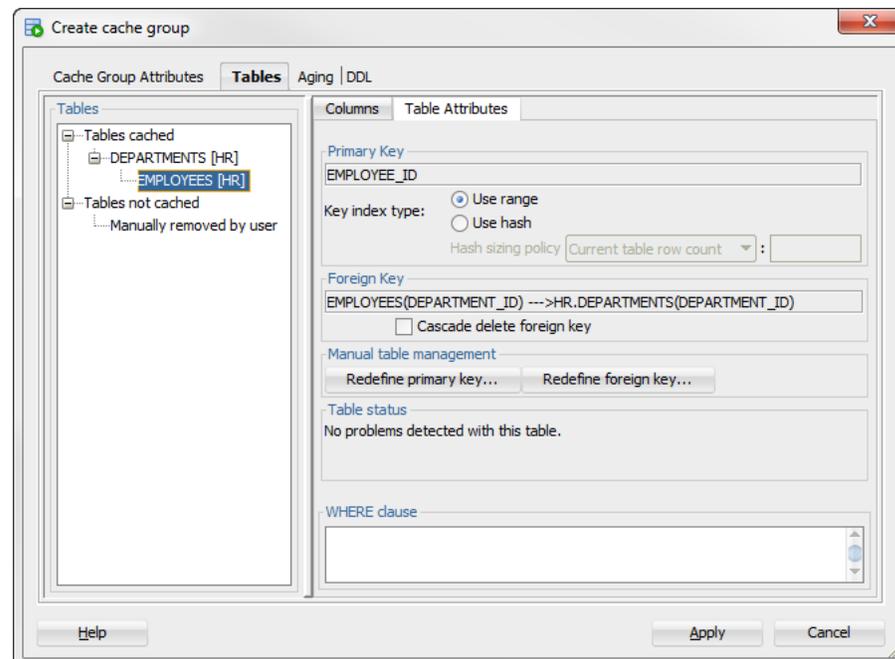
If you would like the foreign key columns of a child table to have cascade delete enabled such that when rows containing referenced key values are deleted from its parent table, rows in the child table with dependent foreign key values are automatically deleted, then select **Cascade delete foreign key**.

By default, the foreign key columns of a child cache table are identical to the foreign key columns of the corresponding Oracle database table. You can define a different column or set of columns as the foreign key for the cache table.

In the **Manual table management** section of the **Table Attributes** tab within the **Tables** tab of the **Create cache group** dialog, select **Redefine foreign key**. See Figure 9–15, "Specifying attributes for child table".

In the **Define Foreign Key(s)** dialog, the columns in the child table whose data types are compatible with the primary key columns of the parent table are shown in the **Foreign Key(s)** drop-down select list. Select the columns for the foreign key of the cache table. Click **OK** to define the alternate foreign key for the cache table.

Figure 9–16   Defining foreign keys
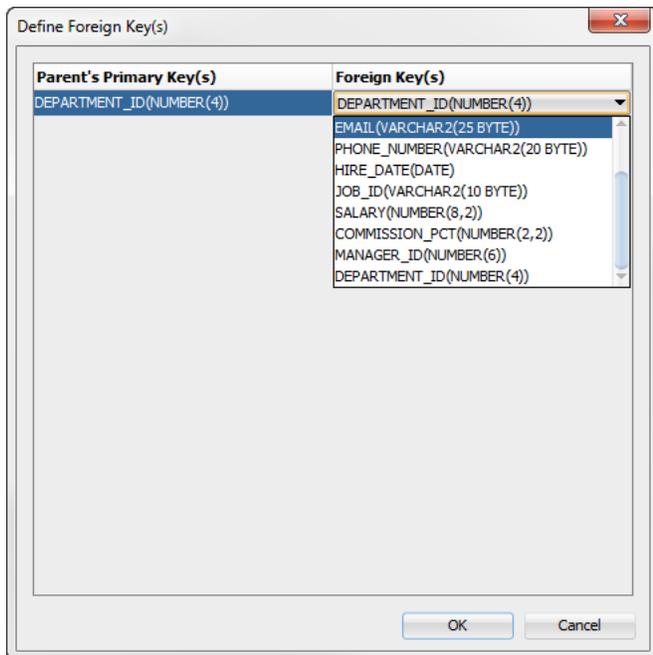
For more information about the hierarchy and relationship of cache tables in a multiple-table cache group, see "Multiple-table cache group" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

If the child cache table is in a user managed cache group that has selected **Varies from table to table** in the **Table attributes** section within the **Cache Group Attributes** tab of the **Create cache group** dialog, then you must specify the table attribute for the cache table in a similar fashion as was specified for the root table. If the cache table is updatable, then specify whether updates on the table are to be propagated or not propagated to the corresponding Oracle database table.

Completing the definitions of the child tables

Define the columns to cache, the type of index to create on the primary key columns, the cascade delete setting for the foreign key columns, and the table attributes including an optional WHERE clause for any remaining child tables.

Verifying the cache group definition

In the **DDL** tab of the **Create cache group** dialog, you can view the CREATE CACHE GROUP statement used to create the cache group and its cache tables.

Click **Apply** to create the cache group and its cache tables.

Figure 9–17   Viewing the cache group definition
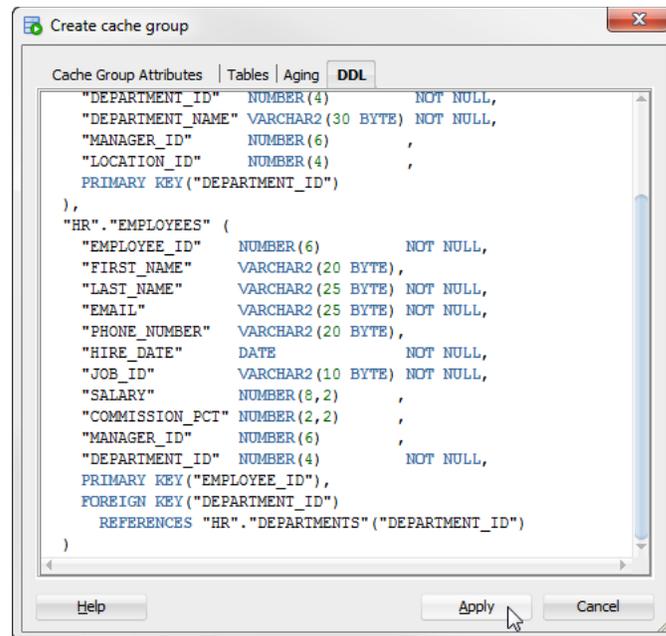
If you created an AWT cache group, then you must start the replication agent for committed updates on its cache tables to be asynchronously propagated to the corresponding Oracle database tables. For information about how to start the replication agent, see "Starting the replication agent" on page 12-3.

Dropping a cache group

When you drop a cache group, its cache tables are also dropped.

Under the **Cache Groups** node within the node of the connection name for the TimesTen database, right-click the name of the cache group to drop and select **Drop**.

Figure 9–18   Drop cache group
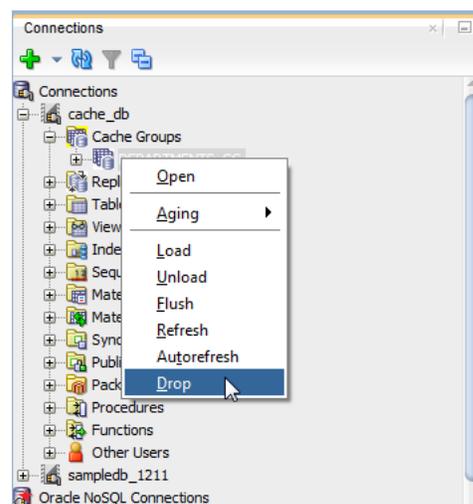
Click **Apply** to drop the cache group.

Figure 9–19   Drop cache group confirmation
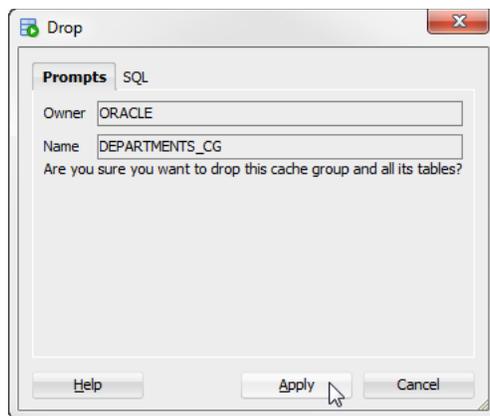
For more information about dropping a cache group, see "Dropping a cache group" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

10

Cache Group Operations

This chapter focuses on the operations you can perform on your cache group. Topics include:

> **Note:** TimesTen cache is not supported with TimesTen Scaleout databases.

- Loading or refreshing a cache group
- Unloading a cache group
- Flushing a cache group

Loading or refreshing a cache group

Click the **+** to the left of the **Cache Groups** node to view the list of cache groups. Next right-click the name of the cache group to load or refresh. Select **Load** to load the cache group or **Refresh** to refresh the cache group.

Figure 10–1   Cache group operations
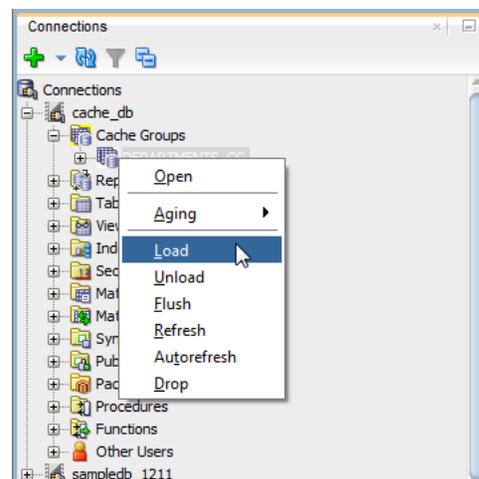
The cache agent must be running to load or refresh a cache group. For information about how to start the cache agent, see "Start and stop the cache agent" on page 8-2.

For a read-only cache group or a user managed cache group that has automatic refresh defined, you must set the automatic refresh state to Paused before you can load or refresh the cache group. For information about changing the automatic refresh state of

a cache group, see "Altering the automatic refresh attributes of a cache group" on page 11-4.

In the **Prompts** tab of the **Load** or **Refresh** dialog, specify a numeric value in the **Commit every n rows** field to indicate the frequency, based on the number of rows that are loaded or refreshed into the cache group, at which a commit is issued during the load or refresh operation. By default, a transaction is committed after every 256 rows are loaded or refreshed into the cache group.

In the **Number of threads to run in parallel** field within the **Prompts** tab of the **Load** or **Refresh** dialog box, specify the number of cache agent process threads to spawn and use for the load or refresh operation. Parallel processing of the load or refresh operation can potentially improve the operation's throughput. The numeric value specified in this field cannot exceed 10 and should not exceed the number of CPUs on the system that can be dedicated to the parallel load or refresh operation. The default value of 1 indicates that the load or refresh operation is processed serially. For more information about parallel loading or refreshing of cache groups, see "Improving the performance of loading or refreshing a large number of cache instances" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

You can specify an optional WHERE clause for the load or refresh operation in the **Where clause** field within the **Prompts** tab of the **Load** or **Refresh** dialog. You can use a WHERE clause for a load or refresh operation on an AWT, SWT or user managed cache group that does not have automatic refresh defined. For more information about specifying a WHERE clause in a load or refresh operation, see "Loading and refreshing a cache group" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

Click **Apply** to load or refresh the cache group.

For more information about loading or refreshing a cache group, see "Loading and refreshing a cache group" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

Unloading a cache group

Click the **+** to the left of the **Cache Groups** node to view the list of cache groups. Next right-click the name of the cache group to unload. Select **Unload** to unload the cache group. See Figure 10–1, "Cache group operations".

In the **Where clause** field within the **Prompts** tab of the **Unload** dialog, specify an optional WHERE clause for the unload operation. You can use a WHERE clause for an unload operation on an AWT, SWT or user managed cache group that does not have automatic refresh defined.

Click **Apply** to unload the cache group.

Figure 10–2   Unload/refresh with optional where clause
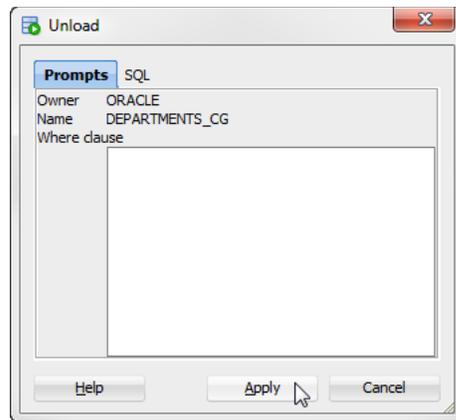
For more information about unloading a cache group, see "Unloading a cache group" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

Flushing a cache group

Only user managed cache groups containing cache tables that are not all read-only, or not all updatable where updates on the tables are propagated to the corresponding Oracle tables can be flushed.

Click the **+** to the left of the **Cache Groups** node to view the list of cache groups. Next right-click the name of the cache group that you to flush. Select **Flush** to flush the cache group. See Figure 10–1, "Cache group operations".

In the **Where clause** field within the **Prompts** tab of the **Flush** dialog, specify an optional WHERE clause for the flush operation.

Click **Apply** to flush the cache group. See Figure 10–2, "Unload/refresh with optional where clause".

For more information about flushing a cache group, see "Flushing a user managed cache group" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

11

Cache Group and Cache Table Characteristics

This chapter focuses on viewing cache table characteristics. After you create a cache group, you can view the characteristics of the cache group including the type of cache group, the automatic refresh attributes and the aging policy (if any). You can also alter the automatic refresh attributes.

> **Note:** TimesTen cache is not supported with TimesTen Scaleout databases.

Topics include:

- Viewing the characteristics of a cache group including:
    - Viewing the cache tables
    - Viewing the cache group definition
    - Viewing the aging attributes
- Altering the automatic refresh attributes of a cache group

Viewing the characteristics of a cache group

Click the **+** to the left of the **Cache Groups** node to view the list of cache groups. Then click the name of the cache group.

Figure 11–1 List of cache groups
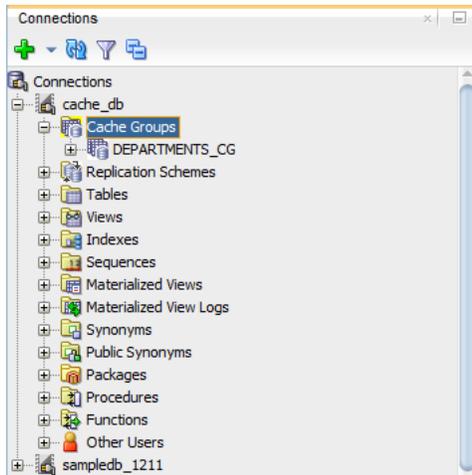
The **Cache Detail** tab shows information such as the cache group type, the name of the root table, and the number of tables in the cache group.

For read-only cache groups and user managed cache groups that have automatic refresh defined, the automatic refresh mode, interval and state is displayed.

Figure 11–2 Viewing the cache group in detail
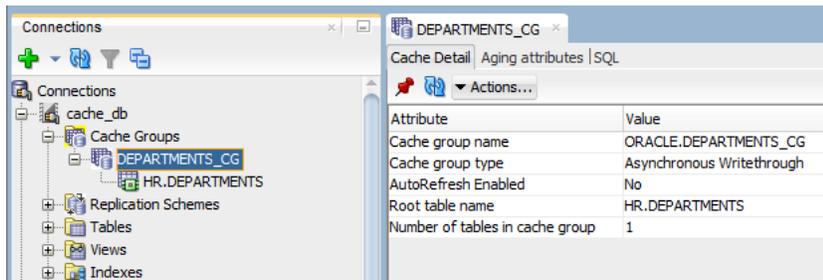
Viewing the cache tables

To view the list of cache tables in a cache group, right-click the **Cache Groups** node, click the **+** to the left of the node that contains the name of the cache group.

Figure 11–3   Viewing cache group tables
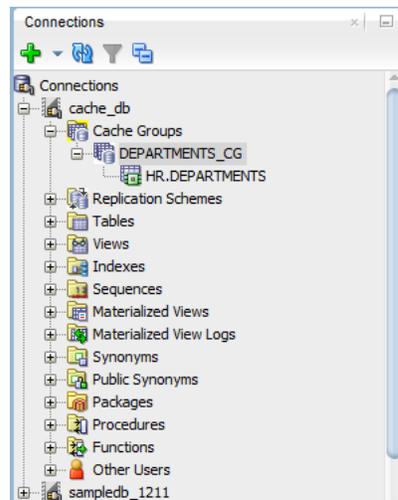
Viewing the cache group definition

To view the CREATE CACHE GROUP statement, select the **cache group** tab and then select the **SQL** tab.

Figure 11–4   Viewing the cache group definition
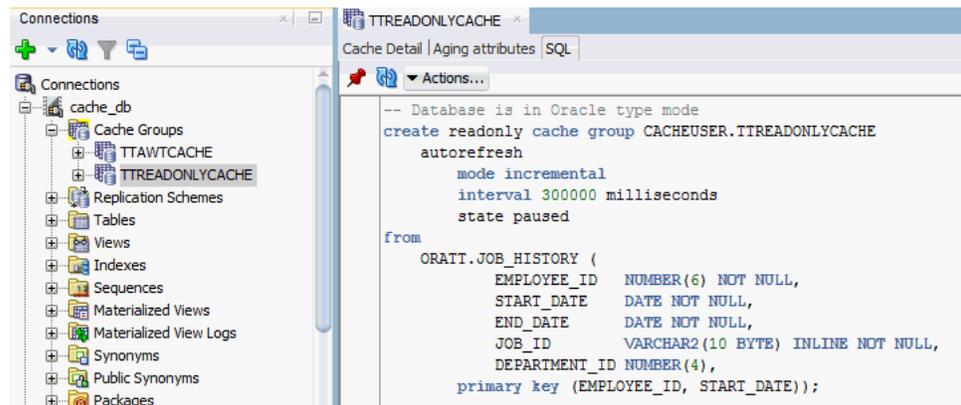
If you are unable to find the cache group, see "Locating a TimesTen database object" on page 4-52 for information.

Viewing the aging attributes

The **Aging attributes** tab, located within the cache group tab, shows information such as the aging policy type, aging cycle, and aging state for cache groups that have an aging policy defined.

For cache groups that have an LRU aging policy, the TimesTen database memory usage thresholds are displayed. For information on LRU aging and LRU thresholds, see "Specifying an aging policy" on page 9-8.

For cache groups that have a time-based aging policy, the name of the non-nullable DATE or TIMESTAMP column used to store the timestamp value (indicating when each row was added or most recently updated) and the length of time that non-updated data is not deleted is displayed.

Figure 11–5   Viewing the aging attributes
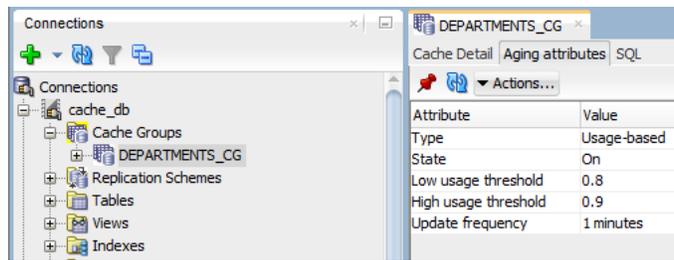
Altering the automatic refresh attributes of a cache group

You can alter the automatic refresh attributes of a read-only cache group or a user managed cache group that has automatic refresh defined.

Under the **Cache Groups** node, right-click the name of the cache group and select **Autorefresh**. See Figure 10–1, "Cache group operations".

In the **Prompts** tab of the **Autorefresh** dialog, you can change any of the following automatic refresh attributes:

■   To change the automatic refresh mode, in the **Mode** field, choose either **Incremental** or **Full**.

■   To change the automatic refresh interval, in the **Refresh Interval** field, specify a numeric interval and units of time in minutes, seconds or milliseconds.

■   To change the automatic refresh state, in the **Refresh State** field, choose **Paused**, **On** or **Off**.

Click **Apply**.

Figure 11–6   Viewing the autorefresh cache options
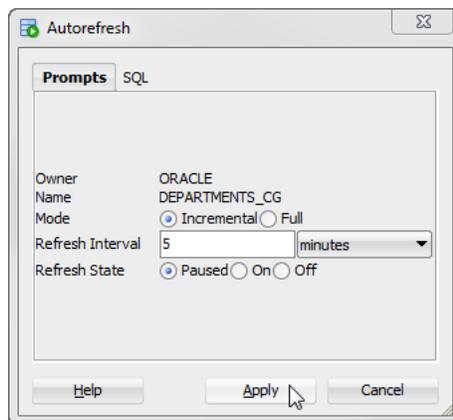
For more information about automatic refresh of cache groups, see "AUTOREFRESH cache group attribute" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

12

Working with TimesTen Replication

A replication scheme defines the set of objects that are replicated between two or more TimesTen databases when committed updates are made on those objects. Replication can be configured at the table or database level in a unidirectional or bidirectional fashion with asynchronous or synchronous transmit.

This chapter focuses on viewing the replication scheme including subscriber databases and `STORE` attributes. The chapter also illustrates how to start the replication agent.

> **Note:** TimesTen replication is not supported with TimesTen Scaleout databases.

Topics include:

- Viewing the replication scheme including:
    - Viewing the details of the TimesTen subscriber databases
    - Viewing the store attributes of the replicated TimesTen databases
- Starting the replication agent

Viewing the replication scheme

To create a replication scheme, you must manually compose and then execute a `CREATE REPLICATION` or `CREATE ACTIVE STANDBY PAIR` statement either outside of SQL Developer using a command-line interface such as the `ttIsql` utility, or within SQL Developer using the **SQL Worksheet**. After you have created a replication scheme, you can view the characteristics of the scheme itself such the database name and host name of the participating TimesTen databases, and its replicated objects.

A replication scheme is automatically created when you create an AWT cache group. This replication scheme is owned by the `TTREP` user.

Click the **+** to the left of the **Replication Schemes** node to view the list of replication schemes. Click the name of the replication scheme to view.

Figure 12–1   Viewing the replication scheme
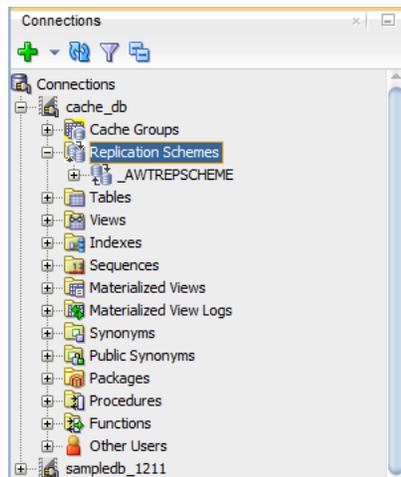
The name of the replication scheme for an active standby pair is `_ACTIVESTANDBY` and it can be found under the **Replication Schemes** node of any existing database user.

The name of the replication scheme that is automatically created to propagate updates on cache tables in AWT cache groups to the corresponding Oracle tables is `_AWTREPSCHEME`.

If you are unable to find the replication scheme that you are looking for, see "Locating a TimesTen database object" on page 4-52.

Viewing the details of the TimesTen subscriber databases

The **Subscriber details** tab, located within the replication scheme tab, displays information such as the database name and host name of the master and subscriber TimesTen databases, the type and database name of the replicated objects, and the return service type.

Figure 12–2   Viewing the subscriber
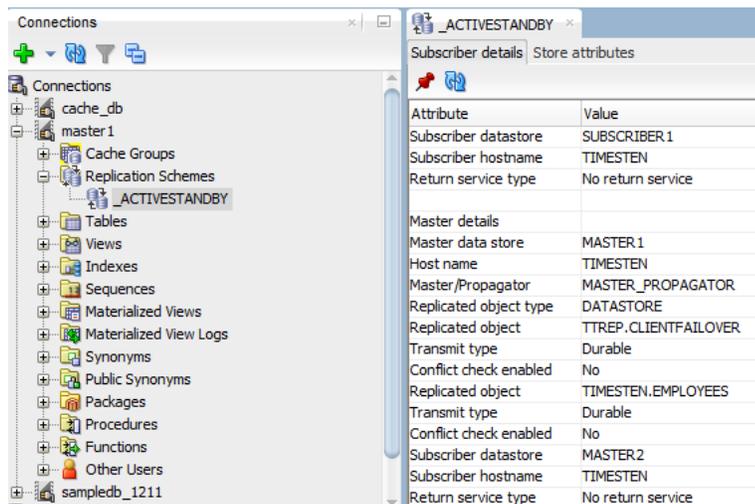
Viewing the store attributes of the replicated TimesTen databases

The **Store attributes** tab, located within the replication scheme tab, displays information such as the TimesTen database name, host name, replication agent port number, replication response timeout period, and transaction log fail threshold for all databases participating in the replication scheme.

For replication schemes that have a return service, the return service failure policy, acknowledgment timeout period, and re-enable policy is displayed.

Figure 12–3   Viewing the store attributes

For more information about the creating a replication scheme, see "Defining Classic Replication Schemes" in the *Oracle TimesTen In-Memory Database Replication Guide*.

Starting the replication agent

The replication agent process transmits committed updates on replicated tables in a master TimesTen database to the corresponding replicated tables in a subscriber TimesTen database. The replication agent is also responsible for asynchronously propagating committed updates on cache tables in AWT cache groups to the corresponding Oracle tables. After you create a replication scheme or an AWT cache group, you must start the replication agent if it is not running.

To start the replication agent, right-click the node of the connection name for the TimesTen database and select **Start/Stop Replication Agent**. See Figure 3–7, "Viewing a TimesTen connection".

The **Starting/stopping the replication agent** dialog appears. In the **Prompts** tab, select **Start agent**. Click **Apply**.

If there are no replication schemes or AWT cache groups in your TimesTen connection, the replication agent is not started.

To stop a running replication agent process, right-click the node of the connection name for the TimesTen database and select **Start/Stop Replication Agent**. See Figure 3–7, "Viewing a TimesTen connection".

The **Starting/stopping the replication agent** dialog appears. In the **Prompts** tab, select **Stop agent**. Click **Apply**.

For more information about the replication agent process, see "Starting and stopping the replication agents" in the *Oracle TimesTen In-Memory Database Replication Guide* or "Managing the replication agent" in the *Oracle TimesTen Application-Tier Database Cache User's Guide*.

13

Working with TimesTen Performance Metrics

TimesTen collects performance metrics from TimesTen system tables, system views, and built-in procedures. SQL Developer allows you to capture snapshots and generate reports that compare two snapshots. A snapshot is a collection of performance metrics.

> **Note:** SQL Developer uses functions from the `TT_STATS` PL/SQL package to capture snapshots and generate reports that compare two snapshots. For more information about the `TT_STATS` PL/SQL package, see "TT_STATS" in the *Oracle TimesTen In-Memory Database PL/SQL Packages Reference*.

Before attempting to use the TimesTen Performance Metrics:

- Ensure that you are using Oracle TimesTen In-Memory Database Release 11.2.2.5 (or later).
- Ensure that the TimesTen connection user has execution privileges on the `TT_STATS` PL/SQL package. Any other user, including an `ADMIN` user, must be granted `EXECUTE` privilege for the `TT_STATS` package by the instance administrator or by an `ADMIN` user, such as in the following example:

```
GRANT EXECUTE ON SYS.TT_STATS TO scott;
```

This chapter provides information about working with TimesTen performance metrics.

Topics include:

- Creating snapshots
- Dropping snapshots
- Editing the TimesTen performance metrics configuration
- Creating a TimesTen performance metrics report

Creating snapshots

A snapshot is a collection of performance metrics from a database. You can create a report comparing two snapshots to analyze the performance of your database.

To create a new snapshot of a database, ensure that you are on the main SQL Developer page and that you have expanded the TimesTen connection in the connections navigator.

1. Right-click the name of the database and select **TimesTen Performance Metrics**, then select **Create Snapshot**.

Figure 13–1   Creating a snapshot
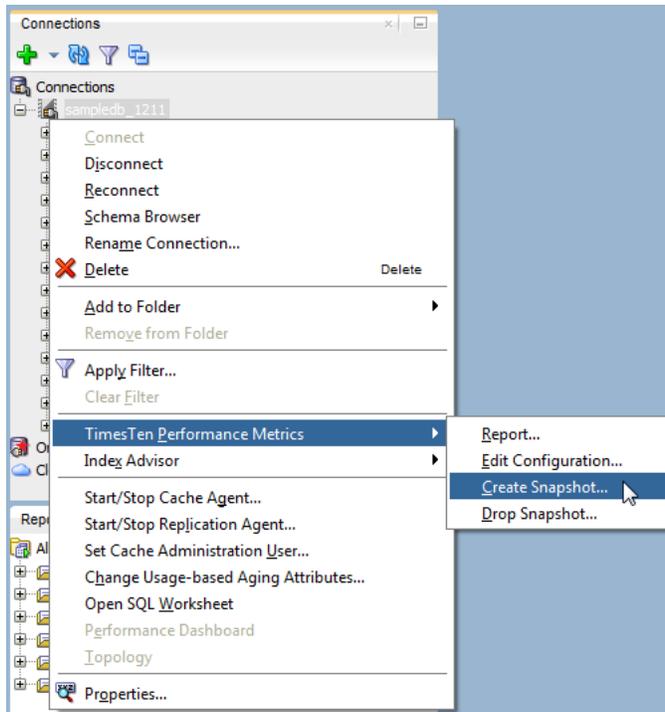
The **New TimesTen Metrics Snapshot** dialog displays.

2. In the **Prompts** tab of the **New TimesTen Metrics Snapshot** dialog, select the desired capture level from the **Capture Level** drop-down list:

    ■   **TYPICAL**: For a typical set of metrics. This level is appropriate for most purposes.

    ■   **NONE**: For metrics outside of SYS.SYSTEMSTATS only.

   ■   **BASIC**: For a minimal basic set of metrics.

    ■   **ALL**: For all available metrics.

    Ensure to use the same capture level for any two snapshots to be used in a report. An error dialog displays if you attempt to generate a report that compares two snapshots with different capture levels.

Figure 13–2   New TimesTen Metrics Snapshot dialog
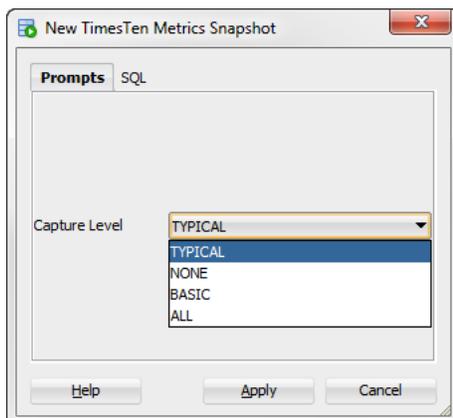
You are now ready to create a new snapshot.

3. Click **Apply** to create the snapshot.

Figure 13–3   Creating new snapshot
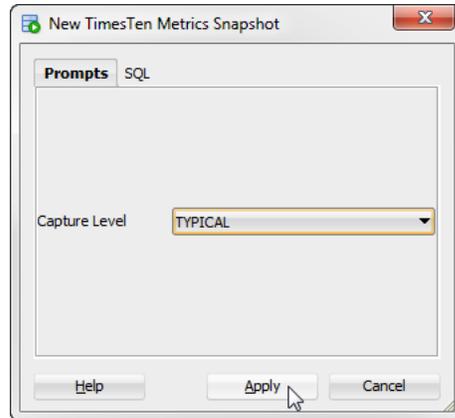
A confirmation dialog shows the ID number of the snapshot. TimesTen generates the snapshot ID.

4. Click **OK**.

Figure 13–4   Snapshot ID confirmation
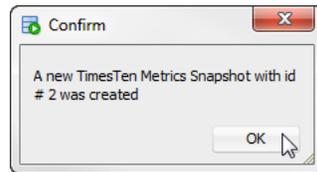
TimesTen creates the snapshot with the specified snapshot ID. The snapshot ID is useful for creating TimesTen metrics reports.

Dropping snapshots

Consider dropping unneeded snapshots to free up permanent space. If you need to save more snapshots of your database, consider editing the TimesTen performance metrics configuration. See "Editing the TimesTen performance metrics configuration" on page 13-5.

To drop a new snapshot from a database, ensure that you are on the main SQL Developer page.

1. Right-click the name of the database and select **TimesTen Performance Metrics**, then select **Drop Snapshot**.

Figure 13–5 Dropping a snapshot
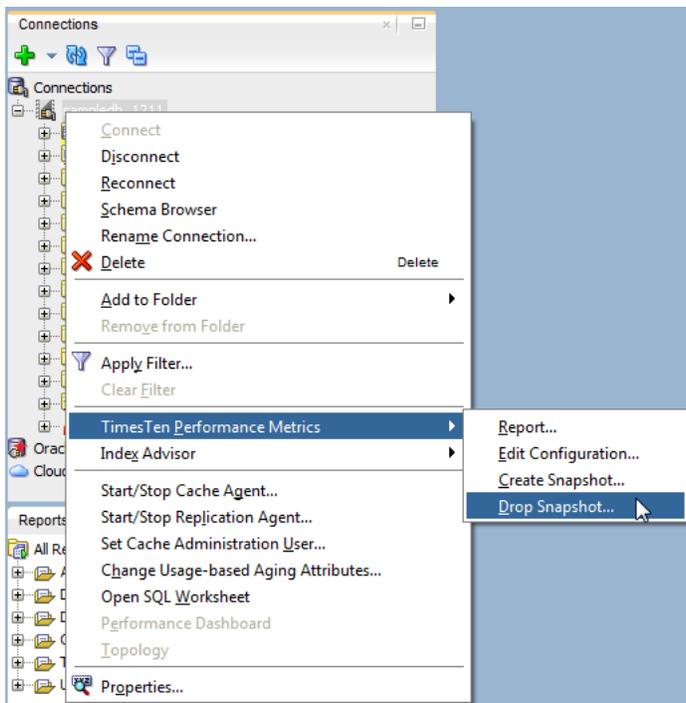
2. By default, the **Prompts** tab of the **Drop TimesTen Metrics Snapshot(s)** dialog is selected. In the **Selector** column, select the snapshots you want to drop.

   You can also click on the column header to sort the table based on the column. A description of each column follows:

   ■ **Selector**

     The check box you select to determine the snapshots that you want to drop.

   ■ **Snapshot**

     The snapshot ID of the snapshot.

   ■ **Timestamp**

     The timestamp of when the snapshot was captured.

   ■ **Capture Level**

     The capture level of the snapshot. This value can be TYPICAL, NONE, BASIC, or ALL.

   To select all snapshots, click the **Select all** check box.

3. Once you have selected the snapshots that you want to drop, click the **Apply** button.

Figure 13–6 Selecting snapshots that you want to drop
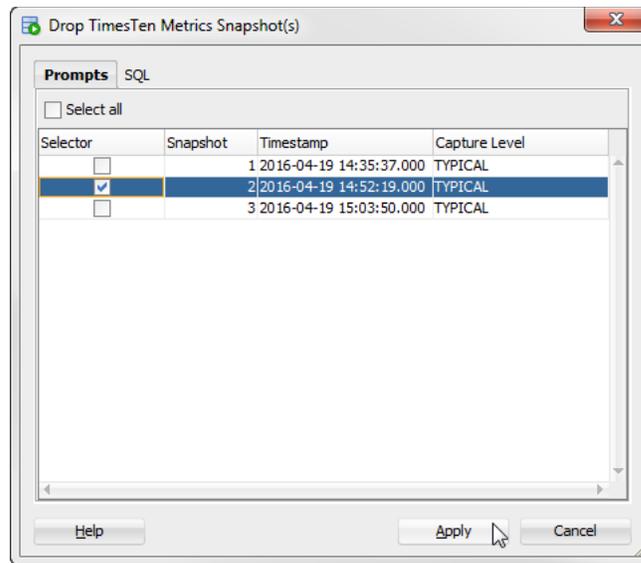
A Confirm dialog displays asking you if you want to proceed with dropping the selected snapshots.

4. Click **Yes**.

Figure 13–7 Confirmation dialog for dropping snapshots
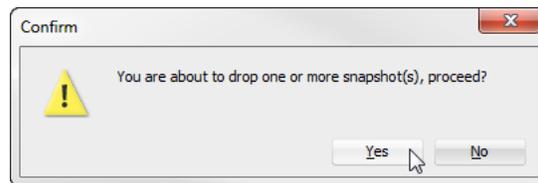
A confirmation dialog displays a message indicating that one or more TimesTen Metrics snapshots were dropped.

5. Click **OK**.

Figure 13–8 Dropped snapshots

You successfully dropped the TimesTen metrics snapshots.

Editing the TimesTen performance metrics configuration

TimesTen performance metrics configuration allows you to modify the maximum snapshot count and the maximum snapshot retention size. These configuration settings are useful in preventing your database from running out of permanent space and in allowing you to save more snapshots.

Once the maximum snapshot count or the maximum snapshot retention size is reached, TimesTen removes the oldest snapshots to free up permanent space. By default, the maximum snapshot count is 256 and the maximum snapshot retention size is 256 MB.

To edit the TimesTen performance metrics configuration for a database, ensure that you are on the main SQL Developer page.

1. Right-click the name of the database and select **TimesTen Performance Metrics**, then select **Edit Configuration**.

Figure 13–9   Editing the TimesTen performance metrics configuration
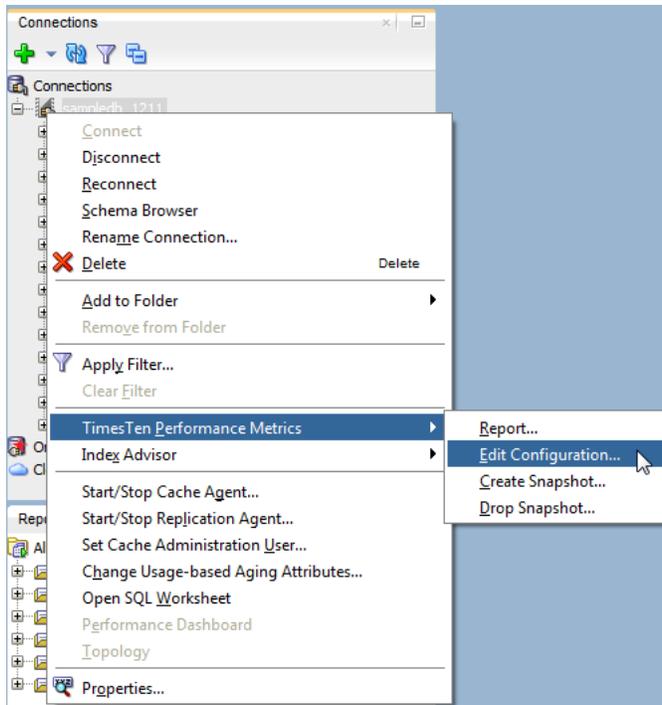
The **TimesTen Metrics Configuration** dialog displays and you see the current configuration for the TimesTen metrics.

2. The **Prompts** tab of the **TimesTen Metrics Configuration** dialog is selected. In the **Maximum Snapshot Count** field, specify a numeric value to indicate the maximum number of snapshots. By default, the maximum number of snapshots is 256.

3. In the **Maximum Snapshot Retention Size (MB)** field, specify the maximum total size of all stored snapshots, in megabytes. By default, the maximum snapshot retention size is 256 MB.

4. Click **Apply**.

Figure 13–10 TimesTen metrics configuration dialog
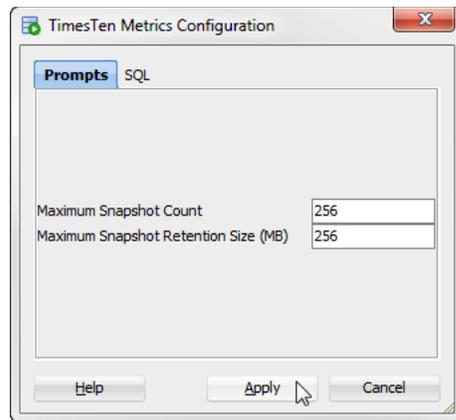
You successfully applied the new TimesTen Metrics configuration settings.

Creating a TimesTen performance metrics report

TimesTen performance metrics reports use the data from two specified snapshots of TimesTen metrics to produce a report in HTML format. TimesTen metrics reports include information for each metric. For more information about the various metrics displayed in the report, see "ttStats" in the *Oracle TimesTen In-Memory Database Reference*.

To generate a new report for a database, ensure that you are on the main SQL Developer page.

1. Right-click the name of the database and select **TimesTen Performance Metrics**, then select **Report**.

Figure 13–11 Creating a report
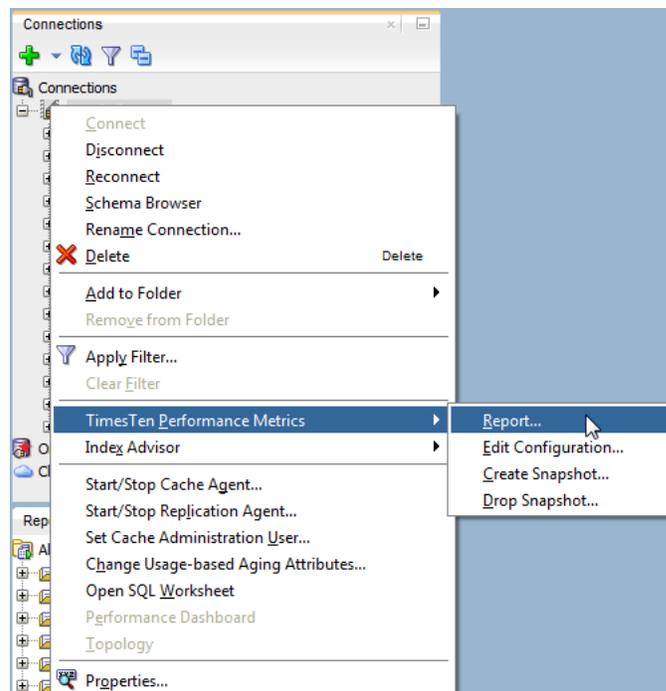
The **Create TimesTen Metrics Report** dialog displays.

2. By default, the **Report generator** tab of the **Create TimesTen Metrics Report** dialog is selected. In the **Selector** column, select two snapshots to compare.

   You can also click on the column header to sort the table based on the column. A description of each column follows:

   - **Selector**

     The check box that determines the snapshots that are compared in the metrics report. Ensure to only select two snapshots with the same capture level. An error dialog displays if you attempt to generate a report that compares more than two snapshots.

   - **Snapshot**

     The snapshot ID of the snapshot.

   - **Timestamp**

     The timestamp of when the snapshot was captured.

   - **Capture Level**

     The capture level of the snapshot. This value can be TYPICAL, NONE, BASIC, or ALL.

3. Once you have selected two snapshots with the same capture level, click the **Generate report** button. An error dialog displays if you attempt to generate a report that compares snapshots with different capture levels.

Figure 13–12   Create TimesTen metrics report
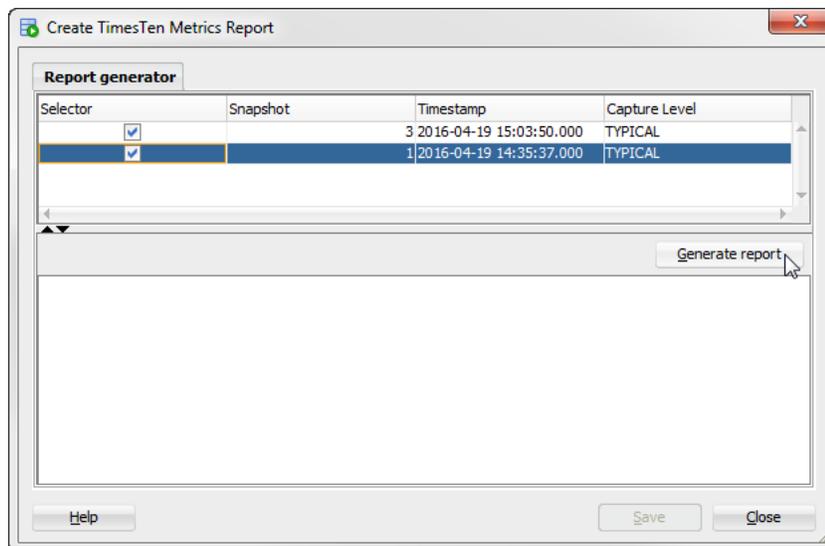
TimesTen generates your TimesTen metrics report. You can view the generated TimesTen metrics report in the pane located below the **Generate report** button.

For more information about the various metrics displayed in the report, see "ttStats" in the *Oracle TimesTen In-Memory Database Reference*.

4. If you do not want to save the report, click **Close**. To save the report, see "Saving a report" on page 13-9.

Saving a report

To save the report for future reference, follow these steps:

1. Click **Save**.

   The Save dialog displays.

2. Select the directory where you want to save the TimesTen metrics report.

3. In the **File Name** field, define the name of your TimesTen metrics report.

   The default filename of the TimesTen metrics report is `ttstats-snapshot_id_1-snapshot_id_2-YYYYMMDDHHMISS.html`, where `snapshot_id_1` and `snapshot_id_2` are the snapshot IDs of the two snapshots you are comparing in your report. `YYYYMMDDHHMISS` is a timestamp of when the report was generated, where `YYYY` is the year, `MM` the month, `DD` the day, `HH` the hour, `MI` the minutes, and `SS` the seconds.

4. Click **Save**.

Figure 13–13   Save TimesTen metrics report
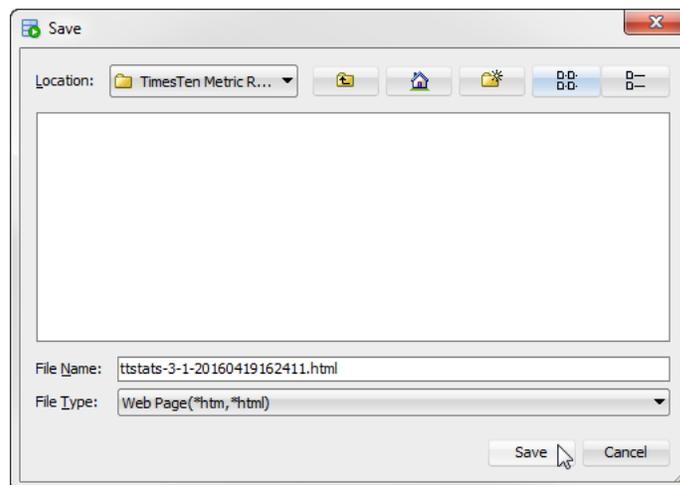
SQL Developer saves your report.

14

TimesTen Reports

You can generate reports about your database and its objects. You can also create your own user-defined reports.

This chapter provides information about generating both TimesTen supplied reports and user-defined reports.

Generating TimesTen Reports

To generate or create reports, navigate to the **Reports** tab below the connections tab in SQL Developer. If this tab is not visible, select **View** and then **Reports**. For more information about how SQL Developer generates and creates reports, see the *Oracle SQL Developer User's Guide*.

You can either choose to create your own user-defined reports or you can choose to generate TimesTen supplied reports.

To create your own user-defined reports, right-click the **User Defined Reports** node and choose **New Report**.

To view the list of TimesTen supplied reports, click the **+** sign to the left of the **TimesTen Reports** node.

TimesTen reports are grouped into the categories:

- About Your Database
- All Objects
- Cache Groups
- Database Administration
- PLSQL
- Security
- Table

About Your Database

There are two reports you can generate:

- **TimesTen configuration**
- **Version Banner**

Choose **TimesTen configuration** to generate a report on the attribute settings for your current database connection. For detailed information on these attributes, see the

built-in procedure "ttConfiguration" in the *Oracle TimesTen In-Memory Database Reference*.

Choose **Version Banner** to generate a report on your database release. For more information, see the built-in procedure "ttVersion" in the *Oracle TimesTen In-Memory Database Reference*.

All Objects

The **All Objects** reports list information about database objects accessible to the user connected to the TimesTen database. There are three types of reports:

- **All Objects**
- **Object Count by Type**
- **Public Synonyms**

Choose **All Objects** to generate a report for each database object accessible to the user. This report includes information on the owner of the object, the name of the object, the type of the object (for example, table, view), the date the object was created, and the date when the last DDL operation was performed on it. The date of the last DDL operation is useful in determining changes to the definition of the object.

Choose **Object Count by Type** to generate a report on the count of all database objects by owner and type.

Choose **Public Synonyms** to generate a report on the names of public synonyms and underlying object type for the public synonyms.

Cache Groups

Cache Groups reports include information about the cache groups owned by the user associated with the specified connection. The reports include:

- **Cache Group Details**
- **User Cache Group Count by Type**

Choose **Cache Group Details** to view reports for the cache groups:

- **Asynchronous Writethrough Cache Groups**
- **Readonly Cache Groups**
- **Synchronous Writethrough Cache Groups**
- **User Managed Cache Groups**

Choose **User Cache Group Count by Type** to view a report that shows the type of cache group and the count for each type of cache group.

Database Administration

Database Administration reports include information on:

- **All Tables**
- **Users**

Choose **All Tables** to generate reports on the names of tables, columns, constraints, and indexes accessible to the user connected to the TimesTen database.

Choose **Users** to generate a report on all internal and external database users.

Performance

Performance reports include information on:

- **Statistics Configuration**
- **Statistics Report**
- **Statistics Snapshots**

Choose **Statistics Configuration** to generate a report on the current statistics configuration of the TimesTen database.

Choose **Statistics Report** to generate an HTML report comparing the two most recent statistics snapshots with the same capture level.

Choose **Statistics Snapshots** to generate a report of all the currently available statistics snapshots. This report includes information on the ID of the snapshot, timestamp of when the snapshot was taken, and the capture level of the snapshot.

For more information on the performance metrics, see Chapter 13, "Working with TimesTen Performance Metrics".

PLSQL

There are three reports you can generate:

- **Program Unit Arguments**
- **Search Source Code**
- **Unit Line Counts**

These reports are generated for PL/SQL procedures, functions, and packages.

Security

Security reports list information about system and object privileges granted to the user connected to the TimesTen database. **Security** reports also list information about the privileges granted to the PUBLIC role.

Security reports are grouped into two categories:

- **Grants and Privileges**
- **Public Grants**

Choose **Grants and Privileges** reports to view information about the system and object privileges granted to the user connected to the TimesTen database. Choose **Public Grants** to view information about the privileges granted to the PUBLIC role. **Public Grants** reports include information on:

- **All Grants**
- **Counts by Grantor**
- **Grant EXECUTE**
- **Grant SELECT**

Table

Table reports include information on:

- **Columns**

- **Constraints**

- **Indexes**

- **Table Sizes**

- **User Synonyms**

- **User Tables**

Choose the **Table** reports to generate reports on the tables, columns, constraints, and indexes owned by the user connected to the TimesTen database. In addition, choose the **Table** reports to generate a report on private synonyms.

Reports for Table Sizes are displayed if you are using TimesTen Release 11.2.2 or later. If you are not using TimesTen Release 11.2.2 or later, then if you choose **Table Sizes**, a **No Query Found** dialog appears. You will see the error: No query found for version of selected database.

Index