Oracle® SQL Developer TimesTen In-Memory Database Support

User's Guide

Release 2.1

E13078-01

November 2009

ORACLE®

Oracle SQL Developer TimesTen In-Memory Database Support User's Guide, Release 2.1

E13078-01

Copyright © 2009, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

Preface v

Audience v
Release Support v
Related Documents vi
Documentation Accessibility vi
Conventions vi

1 Connecting to a TimesTen Database

Set environment variables for TimesTen and SQL Developer 1-1
Configure the Oracle and TimesTen databases 1-2
Define a TimesTen database named connection 1-3
Use an existing TimesTen database named connection 1-5
Disconnect an existing TimesTen database connection 1-6

2 Prerequisites to Creating and Accessing a Cache Environment

Set the cache administration user name and password 2-1
Configure a cache grid 2-2
Start and stop the cache agent 2-3

3 Creating Cache Groups

Create a cache group 3-1
Define cache group and cache table attributes 3-3
Define the root table 3-4
Select the columns to cache 3-5
Specify the table attributes 3-6
Specify an aging policy 3-8
Verify the root table definition 3-10
Define the child tables 3-10
Select the columns to cache 3-11
Specify the table attributes 3-12
Complete the definitions of the child tables 3-14
Verify the cache group definition 3-14
Create an index on a table 3-15
Grant and revoke object privileges to and from database elements 3-18
Drop a cache group 3-20

4 Cache Group Operations

Load or refresh a cache group 4-1
Unload a cache group 4-4
Flush a cache group 4-5

5 Cache Group and Cache Table Characteristics

View the characteristics of a cache group 5-1
View the aging attributes 5-2
View the cache tables 5-3
View the cache group definition 5-4
Locate a TimesTen database element 5-4
View the characteristics of a table 5-6
View the columns 5-7
View the indexes 5-8
View the aging attributes 5-8
View the table definition 5-9
View the privileges granted on a database element 5-10
Alter the automatic refresh attributes of a cache group 5-10
Alter the aging properties of a table 5-12
Add an aging policy to a table 5-12
Drop an aging policy from a table 5-15
Change the aging state for a table's aging policy 5-16
Change the memory usage thresholds and LRU aging cycle 5-17
Change the lifetime and cycle for a table's time-based aging policy 5-18

6 Interfacing with TimesTen Replication

View the characteristics of a replication scheme 6-1
View the details of the TimesTen subscriber databases 6-2
View the store attributes of the replicated TimesTen databases 6-3
Start the replication agent 6-4

7 Interfacing with the TimesTen Query Optimizer

Update the table and column statistics used by the query optimizer 7-1
View the execution plan of a statement 7-4

Preface

This guide is a supplement to the *Oracle Database SQL Developer User's Guide* for TimesTen-specific features.

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 elements; run SQL statements and scripts; manipulate and export data; and view and create reports.

You can connect to a target TimesTen database and perform operations on particular elements in the database.

Audience

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

Release Support

Oracle SQL Developer Release 2.1 supports Oracle TimesTen In-Memory Database Releases 7.0.5 and later, and Oracle TimesTen In-Memory Database and Oracle In-Memory Database Cache Releases 11.2.1.2.0 and later on Microsoft Windows and Linux 32-bit and 64-bit platforms that are supported by the TimesTen software. For information about SQL Developer system recommendations for resources such as CPU, memory and disk storage, see "SQL Developer System Recommendations" in the *Oracle Database SQL Developer Installation Guide*.

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

`http://www.oracle.com/technology/products/database/sql_developer/index.html`

The TimesTen software download is also available on OTN at

`http://www.oracle.com/technology/software/products/timesten/index.html`

Related Documents

SQL Developer 2.1 documentation is available on the product distribution media and on the Oracle Database Documentation Library at

http://download.oracle.com/docs/cd/E15846_01/index.htm

TimesTen documentation is available on the product distribution media and on Oracle Technology Network (OTN) at

http://www.oracle.com/technology/documentation/timesten_doc.html

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at http://www.oracle.com/accessibility/.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Deaf/Hard of Hearing Access to Oracle Support Services

To reach Oracle Support Services, use a telecommunications relay service (TRS) to call Oracle Support at 1.800.223.1711. An Oracle Support Services engineer will handle technical issues and provide customer support according to the Oracle service request process. Information about TRS is available at http://www.fcc.gov/cgb/consumerfacts/trs.html, and a list of phone numbers is available at http://www.fcc.gov/cgb/dro/trsphonebk.html.

Conventions

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 or the glossary.
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.

1

Connecting to a TimesTen Database

After you have installed and started SQL Developer as stated in "Installing and Starting SQL Developer" in the *Oracle Database SQL Developer Installation Guide*, you can connect to a TimesTen database after you have defined a data source name (DSN) for the database.

See "Data source names" in the *Oracle TimesTen In-Memory Database Operations Guide* for information about defining a TimesTen database DSN.

See "Define a DSN for the TimesTen database" in the *Oracle In-Memory Database Cache User's Guide* or "Defining DSNs for cached tables" in the *TimesTen Cache Connect to Oracle Guide* for information about defining a DSN for a TimesTen database that will be used to cache data from an Oracle database.

Note: The component that caches Oracle data in a TimesTen database is documented in the *Oracle In-Memory Database Cache User's Guide* for TimesTen 11*g* and the *TimesTen Cache Connect to Oracle Guide* for TimesTen 7.0.

Set 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 TimesTen Client. If SQL Developer will be used to access a local database, SQL Developer must be installed on the same system as the TimesTen Data Manager. If SQL Developer will be used to access a remote database, SQL Developer must be installed on the same system as the TimesTen Client.

Then execute the *TimesTen_install_dir*/bin/ttenv.sh or *TimesTen_install_dir*/bin/ttenv.csh shell script on Linux systems, or the *TimesTen_install_dir*\bin\ttenv.bat batch file on Microsoft Windows systems corresponding to the TimesTen instance that you want to access from SQL Developer. You must execute the shell script or batch file to set the TimesTen environment variables before starting SQL Developer. The shell script or batch file adds or modifies the following environment variables:

- CLASSPATH includes the *TimesTen_install_dir*/lib/ttjdbc6.jar file on Linux systems or the *TimesTen_install_dir*\lib\ttjdbc6.jar file on Microsoft Windows systems. SQL Developer 2.1 requires Java Development Kit (JDK) 6 be installed.

To determine which third party JDBC drivers are currently being used, select **Tools** > **Preferences**. In the **Preferences** dialog box, click the **+** to the left of the **Database** node to expand the node. Select **Third Party JDBC Drivers** to view the path name of all third party JDBC drivers, including the TimesTen JDBC drivers, that SQL Developer is using.

- LD_LIBRARY_PATH includes the *TimesTen_install_dir*/lib directory on Linux systems.
- PATH includes the *TimesTen_install_dir*/bin directory on Linux systems, or the *TimesTen_install_dir*\bin directory on Microsoft Windows systems.

For TimesTen 7.0, ORACLE_HOME must be set to the Oracle client installation directory before starting the TimesTen main daemon process.

Configure the Oracle and TimesTen databases

Before you can cache Oracle data in a TimesTen database, you must perform certain tasks outside of SQL Developer using command-line interfaces such as SQL*Plus and the ttIsql utility, or within SQL Developer using SQL Worksheet. See "Configuring the Oracle database to cache data in TimesTen" and "Configuring a TimesTen database to cache Oracle data" in the *Oracle In-Memory Database Cache User's Guid*e for information regarding the following tasks:

- Create the timesten user, a schema user and the cache administration user in the Oracle database
- Grant these Oracle users the privileges required to perform desired cache operations
- Create the cache manager user and cache table users in the TimesTen database
- Grant these TimesTen users the privileges required to perform desired cache operations

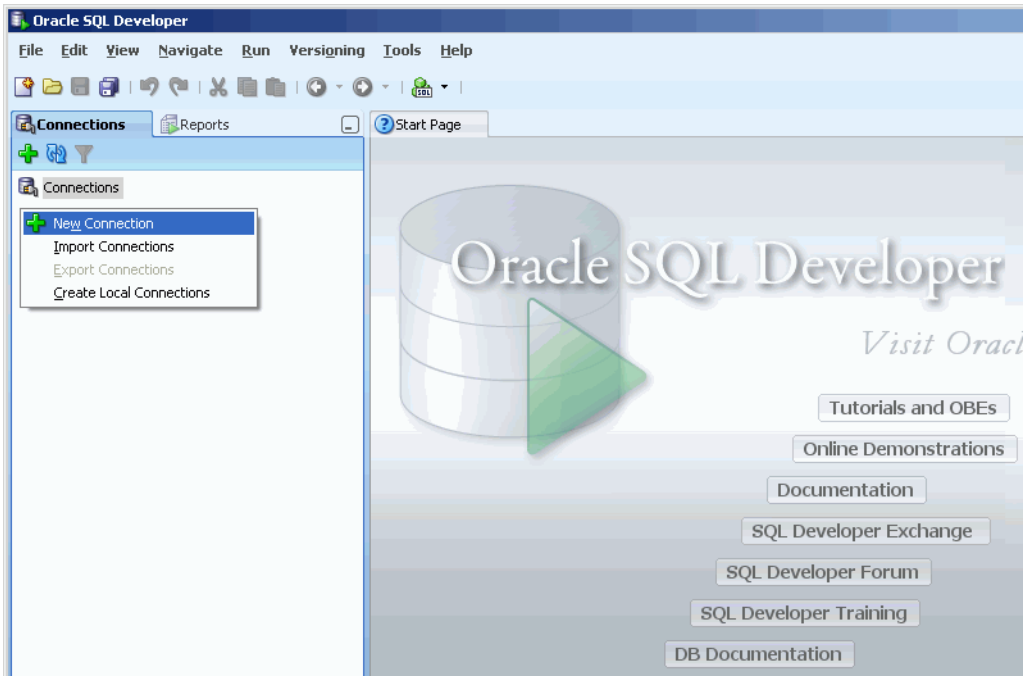
---

Note: These tasks apply only with TimesTen 11*g*.

With TimesTen 7.0, create a schema user and the cache administration user in the Oracle database. Grant these Oracle users the privileges required to perform desired cache operations. Then, if the TimesTen instance has access control enabled, create an internal user in the TimesTen database whose ID is identical to the Oracle schema user. Grant this TimesTen user the privileges required to perform desired cache operations. See the *TimesTen Cache Connect to Oracle Guide* for information regarding these tasks.

Define a TimesTen database named connection

To create a named connection to a TimesTen database, right-click the **Connections** node and select **New Connection**.

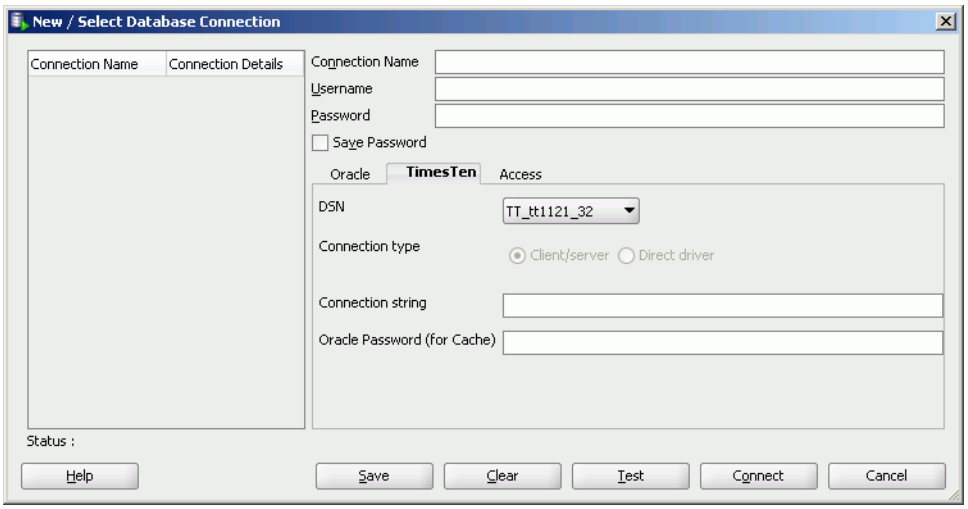


In the **New/Select Database Connection** dialog box, click the **TimesTen** tab and fill in the fields by specifying:

- A connection name in the **Connection Name** field
- A TimesTen user name in the **Username** field. If the TimesTen database will be used to cache data from an Oracle database, specify the TimesTen cache manager user name (or the internal TimesTen user whose ID is identical to the Oracle schema user for TimesTen 7.0) in the **Username** field.
- The password for the user specified within the **Username** field in the **Password** field.

  To save the password in the named connection, put a check in the **Save Password** check box. The **Connection Information** dialog box will not appear and prompt you to enter a password on subsequent connections to the TimesTen database using the named connection.
- If the TimesTen database will be used to cache data from an Oracle database, specify the password of the Oracle user in the **Oracle Password (for Cache)** field. The Oracle user has the same name as the TimesTen user that was specified in the **Username** field.

  Otherwise, leave this field blank.



In the **DSN** pull-down menu, select a predefined DSN that references the TimesTen database that you want to connect to. You can select either a Data Manager DSN to establish a direct connection or a client DSN to establish a client/server connection. If you are going to connect to TimesTen 7.0 and 11*g* databases from the same SQL Developer session using client/server connections, you must have SQL Developer use the TimesTen 11*g* JDBC driver (ttjdbc6.jar file) and not the TimesTen 7.0 driver.

If you selected **User specified** in the **DSN** pull-down menu, select either the **Client/server** button to establish a client/server connection or the **Direct driver** button to establish a direct connection.

See "Configuring TimesTen Client and Server" in the *Oracle TimesTen In-Memory Database Operations Guide* for information about configuring a client/server connection to a TimesTen database.

In the **Connection string** field, specify a connection string to identify the DSN and set DSN attributes if you selected **User specified** in the **DSN** pull-down menu. Each attribute setting must be separated by a semicolon. Do not delimit the connection string with a pair of double quotation marks.

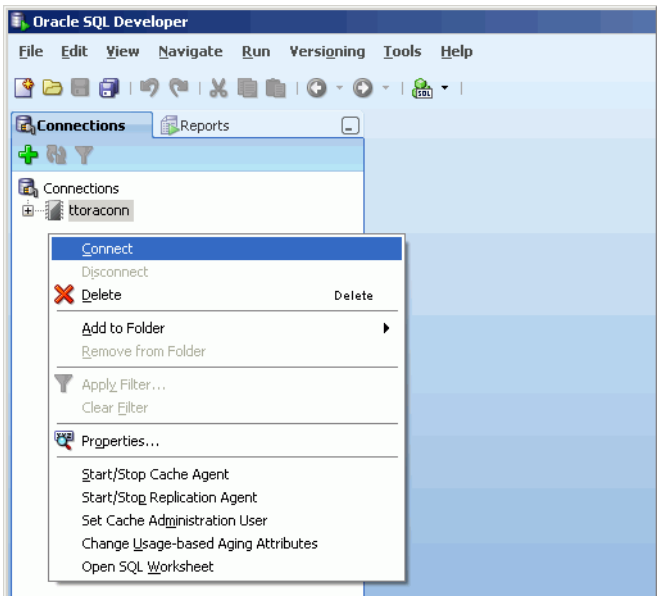
See "Connecting to a data store using a connection string" in the *Oracle TimesTen In-Memory Database Operations Guide* for information about the format and contents of a connection string.

Click the **Save** button to save the connection name under the **Connections** node so that you can establish a connection 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.

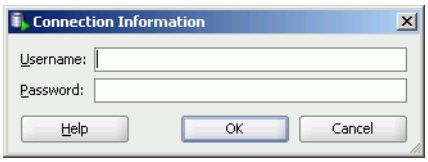
Click the **Connect** button to connect to the TimesTen database.

Use an existing TimesTen database named connection

If a named connection for the TimesTen database that you want to connect to already exists under the **Connections** node, right-click the node that contains the connection name and select **Connect**, or click the **+** to the left of the node that contains the connection name.



If the **Connection Information** dialog box appears, specify a TimesTen user name (the TimesTen cache manager user name if the TimesTen database will be used to cache data from an Oracle database) in the **Username** field, if this field does not already contain a user name, and the corresponding password for this user in the **Password** field. Then click the **OK** button to establish the connection.

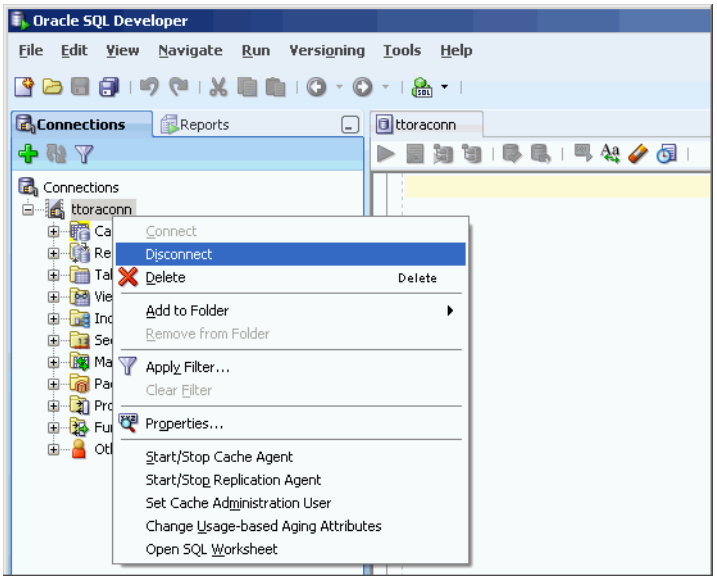


After you connect to a TimesTen database, you can browse the existing elements or create new elements such as tables, indexes, views, sequences, and PL/SQL packages, procedures and functions. See the *Oracle Database SQL Developer User's Guide* for information on creating these elements.

The rest of this guide focuses on working with TimesTen-specific elements such as cache groups, and features such as aging policies.

Disconnect an existing TimesTen database connection

To disconnect an existing connection from a TimesTen database, under the **Connections** node right-click the node that contains the name of the database connection that you want to disconnect and select **Disconnect**.



For more information about creating and managing database connections in SQL Developer, see "Database Connections" in the *Oracle Database SQL Developer User's Guide*.

For information about creating the Oracle and TimesTen users required to cache Oracle data in a TimesTen database, see "Create the Oracle users" and "Create the TimesTen users" in the *Oracle In-Memory Database Cache User's Guide*, or "Create Oracle users and set privileges" and "Create an account on TimesTen" in the *TimesTen Cache Connect to Oracle Guide*.

2

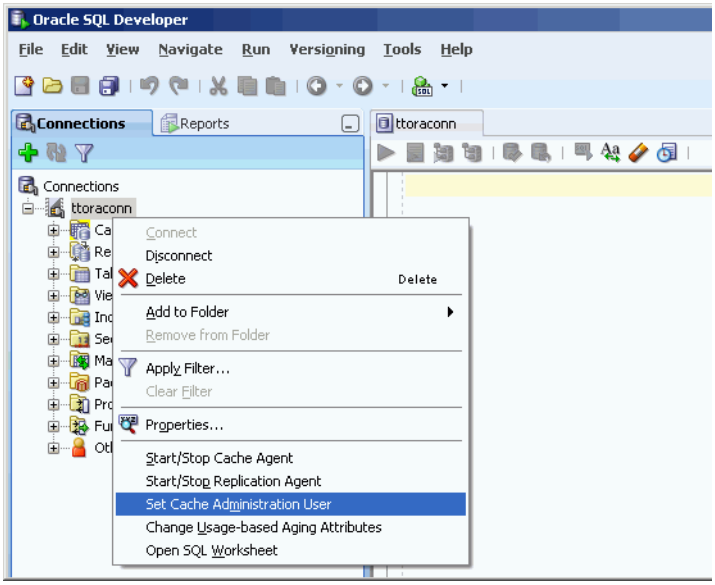
Prerequisites to Creating and Accessing a Cache Environment

After you establish a connection to the TimesTen database, there are some tasks that you must perform before you can start caching Oracle data in the TimesTen database or access cached data.

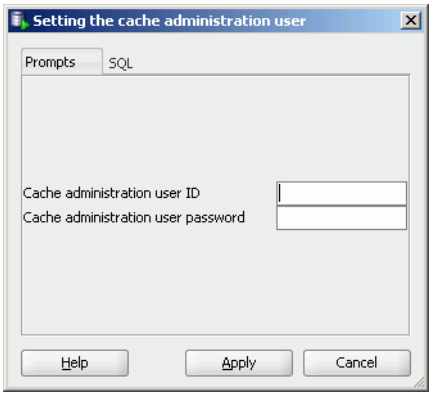
Set the cache administration user name and password

You need to create a cache administration user in the Oracle database. This user creates, owns and maintains Oracle objects that store information used to manage a specific cache grid and enforce predefined behaviors of particular cache group types. After this user is defined, you must set its user name and password in the TimesTen database. (Cache grids apply only to TimesTen 11*g* and later.)

To set the cache administration user name and password, right-click the node that contains the name of your database connection and select **Set Cache Administration User**.



In the **Prompts** tab of the **Setting the cache administration user** dialog box, specify the cache administration user name in the **Cache administration user ID** field, if this field does not already contain a user name, and the corresponding password for this user in the **Cache administration user password** field. Then click the **Apply** button to set the cache administration user name and password in the TimesTen database.



The cache administration user name and password need to be set only once in each TimesTen database that will cache Oracle data unless it needs to be changed. For example, if the TimesTen database is destroyed and re-created, or the cache administration user name is dropped and re-created in the Oracle database, the cache administration user name and password must be set again.

For more information about the cache administration user, see "Set the cache administration user name and password" in the *Oracle In-Memory Database Cache User's Guide* or "Create Oracle users and set privileges" in the *TimesTen Cache Connect to Oracle Guide*.

Configure a cache grid

Before you can create a cache group in a TimesTen database, you must perform certain tasks outside of SQL Developer using a command-line interface such as the ttIsql utility, or within SQL Developer using SQL Worksheet. See "Configuring a cache grid" in the *Oracle In-Memory Database Cache User's Guide* for information regarding the following tasks:

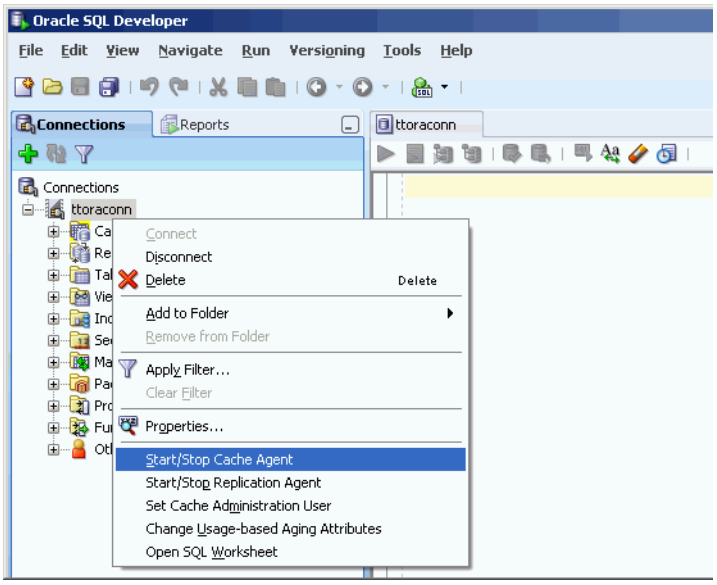
- Create a cache grid
- Associate the TimesTen database with the cache grid

Note: These tasks apply only with TimesTen 11*g* and later.

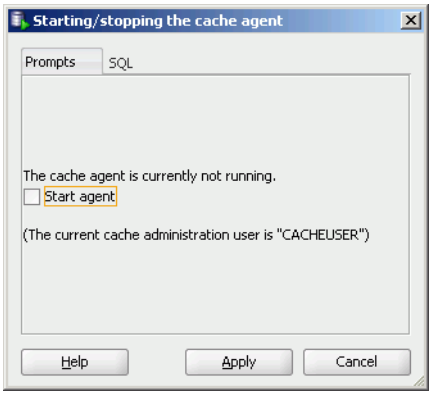
Start and stop the cache agent

After you set the cache administration user name and password, create a cache grid and associate the TimesTen database with the cache grid, you need to start the cache agent process if it is not already running. This TimesTen process performs cache operations such as loading a cache group and automatic refresh.

To start the cache agent, right-click the node that contains the name of your database connection and select **Start/Stop Cache Agent**.

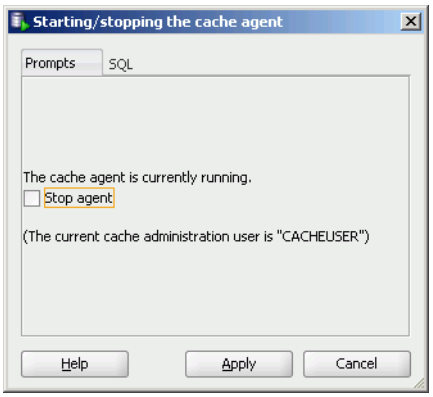


In the **Starting/stopping the cache agent** dialog box, put a check in the **Start agent** check box within the **Prompts** tab. Then click the **Apply** button to start the cache agent.



To stop a running cache agent process, right-click the node that contains the name of your database connection 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, put a check in the **Stop agent** check box within the **Prompts** tab. Then click the **Apply** button to stop the cache agent.



For more information about the cache agent process, see "Managing the cache agent" in the *Oracle In-Memory Database Cache User's Guide* or "Starting and stopping the cache agent" in the *TimesTen Cache Connect to Oracle Guide*.

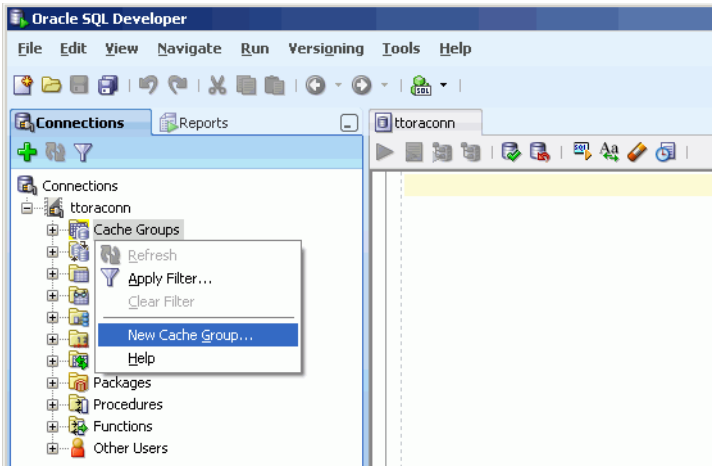
3

Creating Cache Groups

A cache group defines the set of Oracle data to cache in the TimesTen database. When you create a cache group, cache tables are created in the TimesTen database that correspond to the Oracle tables being cached.

Create a cache group

To create a cache group, right-click the **Cache Groups** node under the node that corresponds to the name of your database connection and select **New Cache Group**.



In the **Create cache group** dialog box, specify a name for the cache group in the **Name** field within the **Cache Group Attributes** tab. In the **Type** pull-down menu, select 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, check the **Dynamic** check box in the **Type** section. Otherwise, leave the check box unchecked if you are creating an explicitly loaded cache group.

In an explicitly loaded cache group, data is loaded manually into its cache tables from its corresponding Oracle 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 its corresponding Oracle 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 unless it contains all of the following characteristics:

- Automatic refresh is defined
- One of more of its cache tables are updatable
- Committed updates on at least one of its updatable cache tables will be propagated to its corresponding Oracle tables

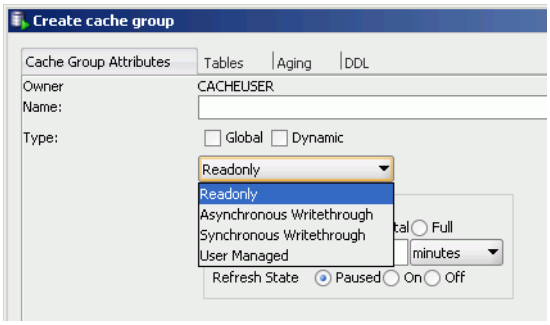
For more information about explicitly loaded and dynamic cache groups, see "Loading data into a cache group: Explicitly loaded and dynamic cache groups" and "Dynamic cache groups" in the *Oracle In-Memory Database Cache User's Guide*.

Cache groups are also classified as either local or global. If you are creating a global cache group, check the **Global** check box in the **Type** section. Otherwise, leave the check box unchecked if you are creating a local cache group.

In a local cache group, data in its cache tables are not shared across TimesTen databases even if the databases are members of the same cache grid. In a global cache group, data in its cache tables are shared across TimesTen databases that are members of the same cache grid.

Only a dynamic AWT cache group can be defined as global cache group.

For more information about local and global cache groups, see "Sharing data across a cache grid: Local and global cache groups" and "Global cache groups" in the *Oracle In-Memory Database Cache User's Guide*.



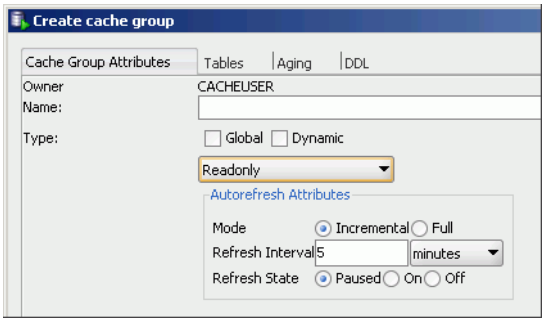
Note: Dynamic and global cache groups are supported in TimesTen 11*g* and later.

All cache groups in TimesTen 7.0 are categorized as explicitly loaded and classified as local. However, you can load data into the cache tables of a cache group from its corresponding Oracle tables on demand using a transparent or dynamic load operation. Transparent load applies to all cache group types except for read-only cache groups and user managed cache groups that have automatic refresh defined. See "Using transparent loading" in the *TimesTen Cache Connect to Oracle Guide* for more information.

Define cache group and cache table attributes

Within the **Cache Group Attributes** tab of the **Create cache group** dialog box, if you are creating a read-only cache group, or a user managed cache group and you put a check in the **Autorefresh** check box within the **Table Attributes** section to enable automatic refresh when either the **All tables readonly** or **Propagate all tables** button is selected, fill in the following automatic refresh characteristics:

- Set the automatic refresh mode in the **Mode** field by selecting either the **Incremental** or **Full** button. 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 from the pull-down menu. The default automatic refresh interval is 5 minutes.
- Set the automatic refresh state in the **Refresh State** field by selecting either the **Paused**, **On** or **Off** button. The default automatic refresh state is Paused.

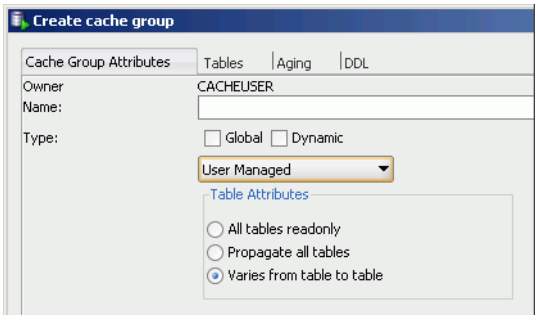


For more information about automatic refresh of cache groups, see "AUTOREFRESH cache group attribute" in the *Oracle In-Memory Database Cache User's Guide* or the *TimesTen Cache Connect to Oracle Guide*.

For more information about read-only cache groups, see "Read-only cache group" in the *Oracle In-Memory Database Cache User's Guide* or "READONLY cache groups" in the *TimesTen Cache Connect to Oracle Guide*.

If you are creating a user managed cache group, select one of the following buttons from the **Table Attributes** section in the **Cache Group Attributes** tab within the **Create cache group** dialog box:

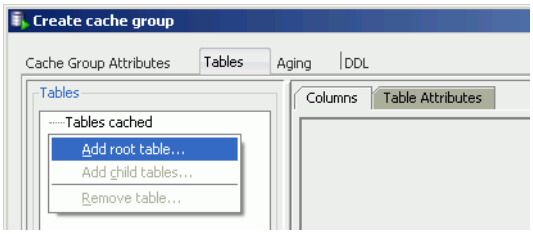
- **All tables readonly**: All cache tables will be read-only
- **Propagate all tables**: All cache tables will be updatable, and committed updates on the cache tables will be propagated to its corresponding Oracle tables
- **Varies from table to table**: Some cache tables will be read-only, and other cache tables will be updatable where committed updates on the cache tables may or may not be propagated to its corresponding Oracle tables



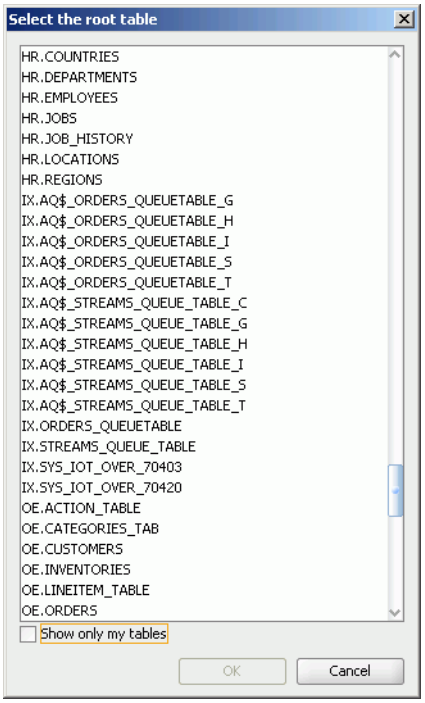
For more information about user managed cache groups, see "User managed cache group" in the *Oracle In-Memory Database Cache User's Guide* or "USERMANAGED cache groups" in the *TimesTen Cache Connect to Oracle Guide*.

Define the root table

Click the **Tables** tab in the **Create cache group** dialog box. Right-click in the **Tables** panel and select **Add root table**.



In the list of Oracle tables shown within the **Select the root table** dialog box, select the table that you want to designate as the root table for your cache group and then click the **OK** button.



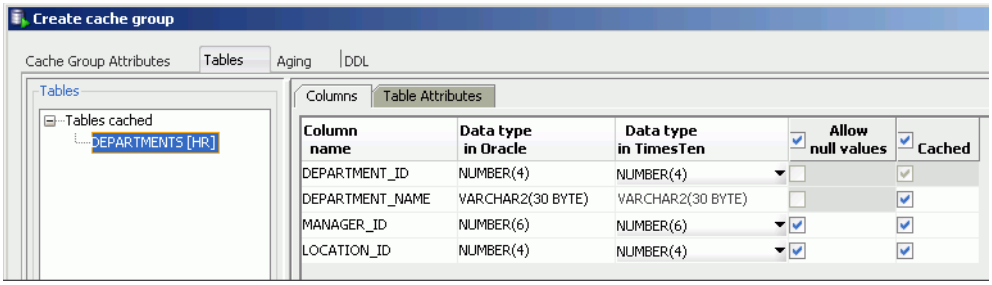
To show only tables owned by the Oracle user who has the same name as the TimesTen cache manager user, put a check in the **Show only my tables** check box.

Select the columns to cache

In the **Columns** tab within the **Tables** tab of the **Create cache group** dialog box, put a check in the check boxes under the **Allow null values** field for each cached column that you want null values to be accepted in. By default, the nullability of a cache table's column is identical to that of the corresponding Oracle table's column. The Oracle table's unique identifying columns cannot accept null values.

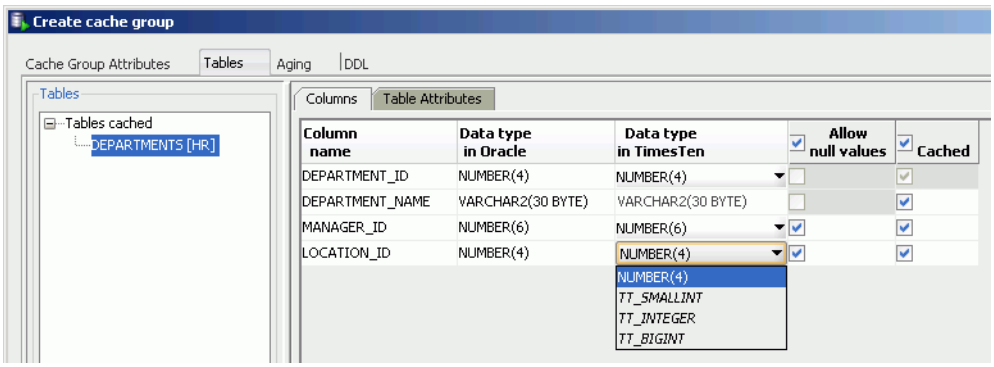
Put a check in the check boxes under the **Cached** field for each column that you want to cache. By default, all columns of the table are selected and will be cached. The Oracle table's unique identifying columns (primary key, unique index or unique constraint) for each row must be included in the TimesTen cache table and will constitute the cache table's primary key.

To select all columns, or deselect all columns except for the Oracle table's unique identifying columns, to accept null values or to cache, put (or do not put) a check in the check boxes that appear to the left of the **Allow null values** and **Cached** fields, respectively.



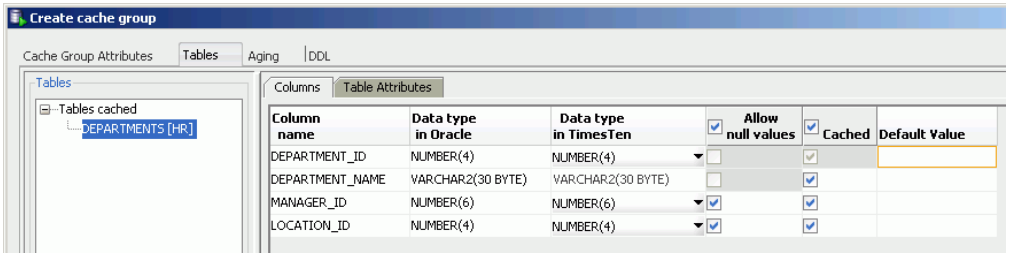
For more information about the characteristics of cache tables and cache groups, see "Cache groups and cache tables" in the *Oracle In-Memory Database Cache User's Guide* or "Defining cache group tables" in the *TimesTen Cache Connect to Oracle Guide*.

Some Oracle data types have more than one compatible TimesTen data type. For columns in the Oracle table with one of these Oracle data types, a pull-down menu appears under the **Data type in TimesTen** field for those columns. The pull-down menu allows you to select the desired data type for the column in the cache table. The non-default compatible or mapped data types appear in italics.



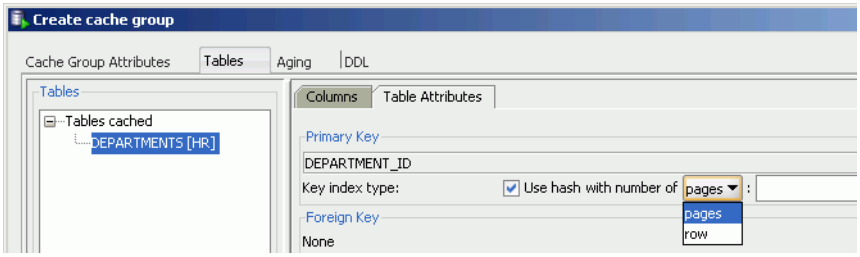
For more information about data type mappings between an Oracle and TimesTen database, see "Mapping between Oracle and TimesTen data types" in the *Oracle In-Memory Database Cache User's Guide* or "Data type mappings for Cache Connect to Oracle" in the *TimesTen Cache Connect to Oracle Guide*.

For updatable cache tables, you can specify a default column value under the **Default Value** field for each cached column that you want a default value defined.



Specify the table attributes

By default, TimesTen creates a range index (called a T-tree index in TimesTen 7.0) 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 the **Create cache group** dialog box, put a check in the **Use hash** check box. You will then need to size the hash index. Specify either the number of pages to allocate to the index or the maximum number of rows the underlying cache table is anticipated to have in the **with number of** pull-down menu and the adjacent field after the colon (:) which requires a numeric value.

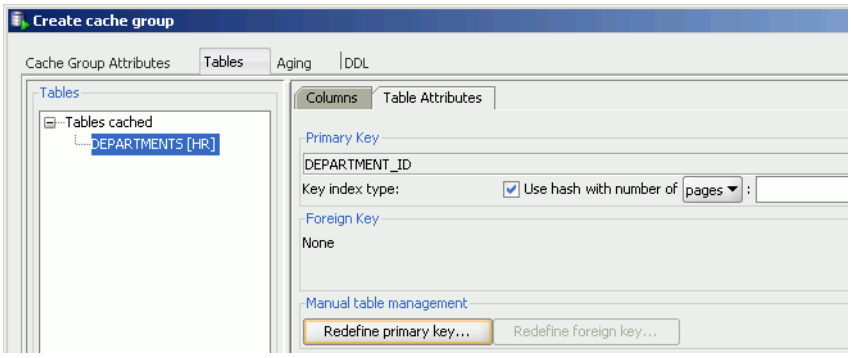


For more information about the index that is automatically created on a cache table's primary key columns, see "CREATE CACHE GROUP" in the *Oracle TimesTen In-Memory Database SQL Reference*.

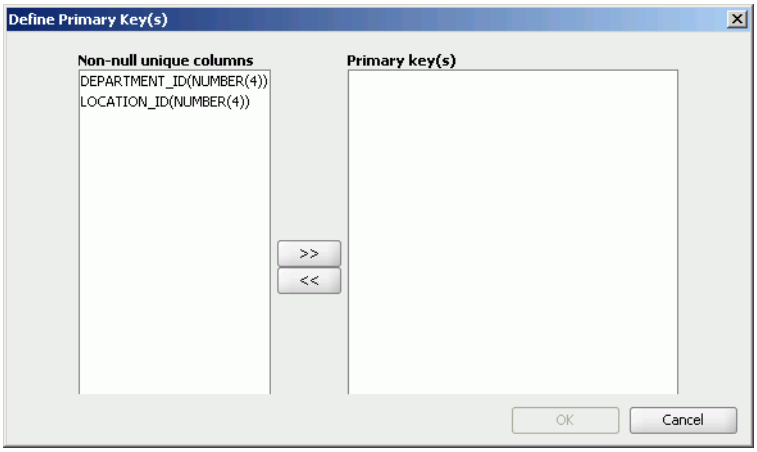
For more information about properly sizing a hash index, see "CREATE TABLE" in the *Oracle TimesTen In-Memory Database SQL Reference*.

By default, the primary key columns of a cache table is the same as the primary key columns of the corresponding Oracle table. You can define a different column or set of columns as the primary key of the cache table.

In the **Manual table management** section of the **Table Attributes** tab within the **Tables** tab of the **Create cache group** dialog box, click the **Redefine primary key** button.



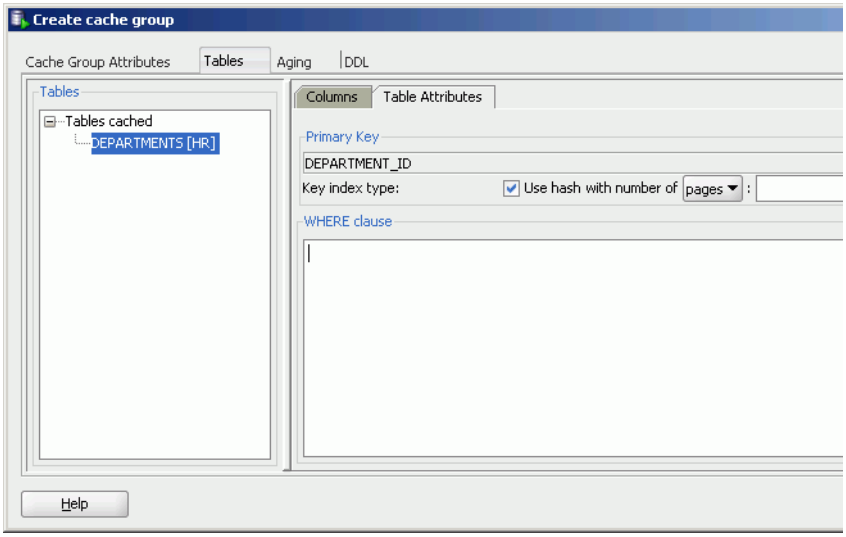
In the **Define Primary Key(s)** dialog box, the non-nullable unique columns of the corresponding Oracle table is shown in the **Non-null unique columns** list. Select the columns that you want the primary key to consist of and then click the **>>** button to move those columns into the **Primary key(s)** list. The order of a composite primary key's columns is significant if a range index is to be created on the columns. Click the **OK** button to define the new set of primary key columns.



If the primary key of the TimesTen cache table is defined on a different set of columns than the primary key of the corresponding cached Oracle table, the tables may become unsynchronized when committed updates are made on the cache table or the corresponding Oracle table. Unique constraint violations can occur when:

- Committed updates on the cache table are propagated to the corresponding Oracle table
- Committed updates on the cached Oracle table are refreshed to the corresponding TimesTen cache table

You can specify an optional WHERE clause for the table in the **WHERE clause** field of the **Table Attributes** tab within the **Tables** tab of the **Create cache group** dialog box. 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.

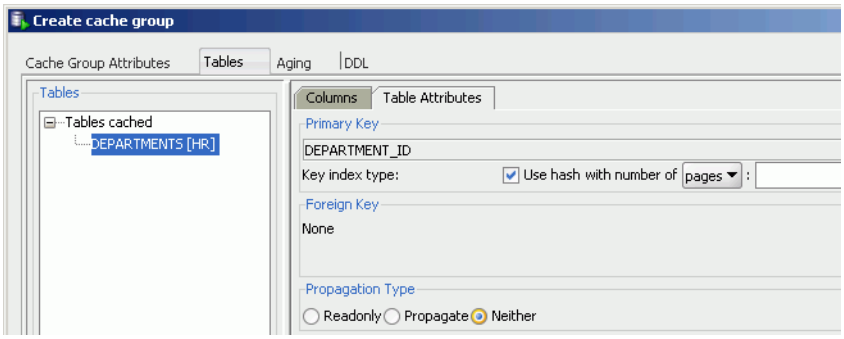


For more information about specifying a WHERE clause in the table definition of a cache group, see "Using a WHERE clause" in the *Oracle In-Memory Database Cache User's Guide* or "Using WHERE clauses" in the *TimesTen Cache Connect to Oracle Guide*.

If the cache table is in a user managed cache group that has selected the **Varies from table to table** button in the **Table attributes** section within the **Cache Group Attributes** tab of the **Create cache group** dialog box, you need to specify the attribute for the table.

In the **Propagation Type** section of the **Table Attributes** tab within the **Tables** tab of the **Create cache group** dialog box, select the **Readonly** button to make the cache table read-only, the **Propagate** button to make the cache table updatable and updates on the table be propagated to its corresponding Oracle table, and the **Neither** button to make the cache table updatable and updates on the table not be propagated to its corresponding Oracle table.

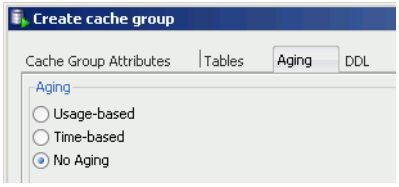
By default, cache tables in a user managed cache group are updatable and updates on the table are not propagated to its corresponding Oracle table.



Specify an aging policy

You can specify an optional aging policy for the cache table in the **Aging** tab of the **Create cache group** dialog box. An aging policy is explicitly defined on the root table of a cache group but it implicitly applies to all its child tables as well.

Select the **Usage-based** button to define a least recently used (LRU) aging policy, the **Time-based** button to define a time-based aging policy, or the **No aging** button to define no aging policy.

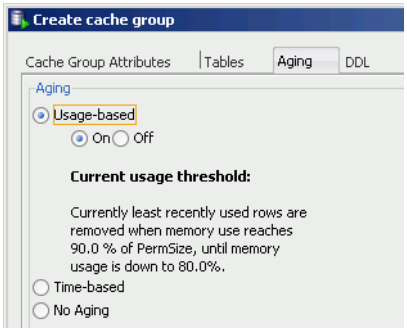


An LRU aging policy can only be defined for tables in an explicitly loaded AWT, SWT or user managed cache group that does not have automatic refresh defined. LRU aging is defined by default for tables in a dynamic cache group.

The TimesTen database memory usage thresholds that determine when data starts and stops being removed from the tables are shown when the **Usage-based** button is selected. The default usage threshold that determines when data starts being removed from the tables is 90%. The default usage threshold that determines when data stops being removed 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 "Change the memory usage thresholds and LRU aging cycle" on page 5-17.

Specify an aging state by selecting either the **on** or **off** button. The default aging state is on.



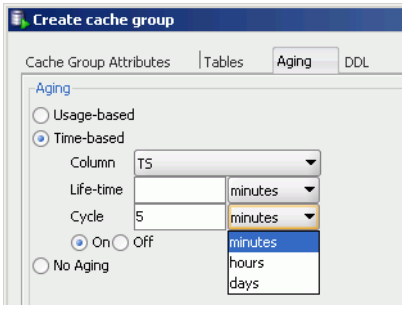
For more information about defining an LRU aging policy on cache tables, see "LRU aging" in the *Oracle In-Memory Database Cache User's Guide* or "Usage-based aging" in the *TimesTen Cache Connect to Oracle Guide*.

A time-based aging policy can only be defined on a cache table that contains a non-nullable DATE or TIMESTAMP column. Columns of these data types for the table will appear in the **Column** pull-down menu within the **Aging** tab of the **Create cache group** dialog box when the **Time-based** button is selected. Select the column that you want to store the timestamp value indicating when each row was inserted or most recently updated.

Indicate the length of time in which data that has not been updated is to be kept in the table by specifying, in the **Life-time** field, a numeric value followed by a unit of minutes, hours or days in the pull-down menu.

Indicate the frequency at which data is to be aged out of the table by specifying, in the **Cycle** field, a numeric value followed by a unit of minutes, hours or days in the pull-down menu. The default time-based aging cycle is 5 minutes.

Specify an aging state by selecting either the **on** or **off** button. The default aging state is on.

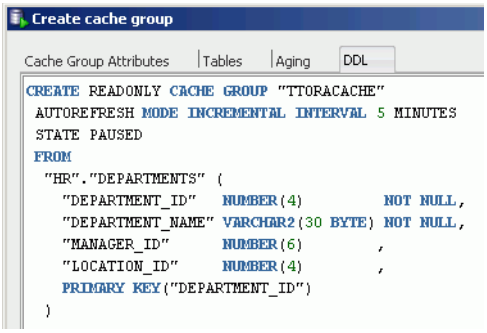


For more information about defining a time-based aging policy on cache tables, see "Time-based aging" in the *Oracle In-Memory Database Cache User's Guide* or the *TimesTen Cache Connect to Oracle Guide*.

Verify the root table definition

After you have defined the tables and columns to cache, 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 in the **DDL** tab of the **Create cache group** dialog box that will be used to create the cache group and its cache table.

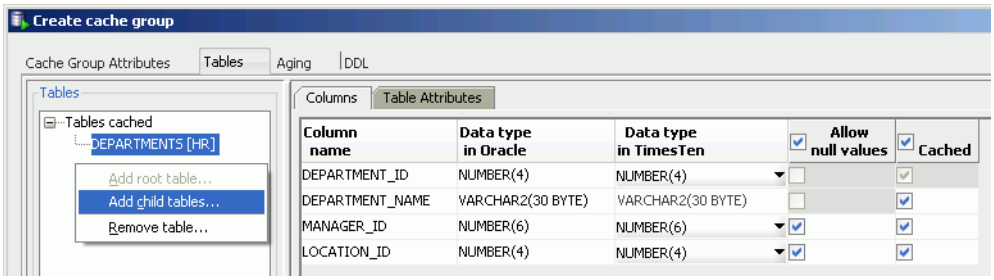
Click the **Apply** button to create the cache group and its cache table if you are not going to include any child tables in the cache group.



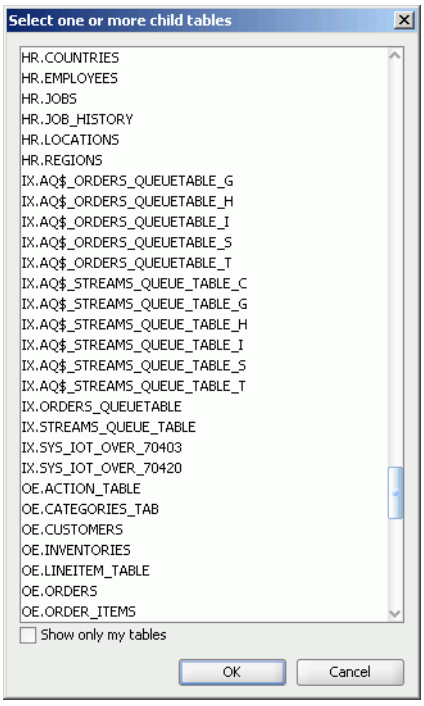
For information about the syntax and semantics of the CREATE CACHE GROUP statement, see "CREATE CACHE GROUP" in the *Oracle TimesTen In-Memory Database SQL Reference*.

Define the child tables

Within the **Tables** tab of the **Create cache group** dialog box, right-click in the **Tables** panel and select **Add child tables**.



In the list of Oracle tables shown within the **Select one or more child tables** dialog box, select the tables that you want to designate as the child tables for your cache group and then click the **OK** button.



To select more than one child table, press and hold the CTRL key, and click the desired tables. The child tables will appear under their appropriate parent table.

Select the columns to cache

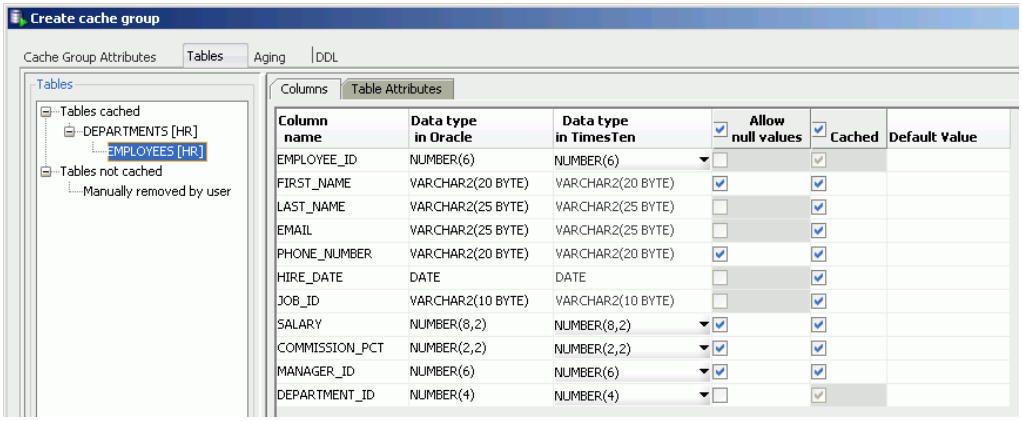
Click a child table and in the **Columns** tab within the **Tables** tab of the **Create cache group** dialog box, put a check in the check boxes under the **Allow null values** field for each cached column that you want null values to be accepted in. By default, the nullability of a cache table's column is identical to that of the corresponding Oracle table's column. The Oracle table's unique identifying columns cannot accept null values.

Put a check in the check boxes under the **Cached** field for each column that you want to cache. By default, all columns of the table are selected and will be cached. The Oracle table's unique identifying columns for each row must be included in the TimesTen cache table and will constitute the cache table's primary key. The Oracle 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 table's unique identifying columns and the foreign key columns that reference its parent table, to accept null values or to cache, put (or do not put) a check in the check boxes that appear to the left of the **Allow null values** and **Cached** fields, respectively.

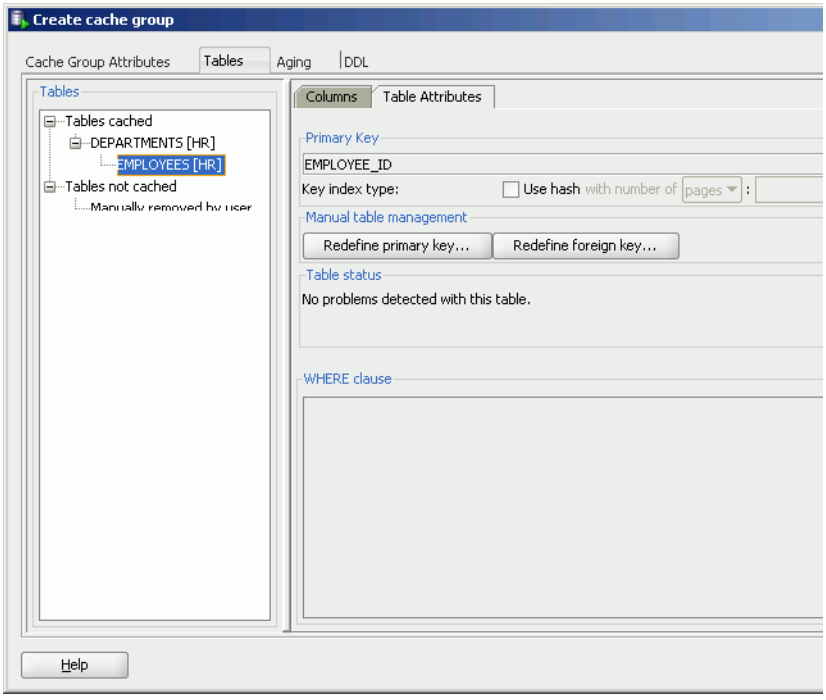
For columns in the Oracle table with more than one compatible TimesTen data type, select the desired data type for the columns in the cache table from the pull-down menu that appears under the **Data Type in TimesTen** field for those columns in a similar fashion as was specified for the root table.

For updatable cache tables, you can specify a default column value under the **Default Value** field for each cached column that you want a default value defined in a similar fashion as was specified for the root table.



Specify the table attributes

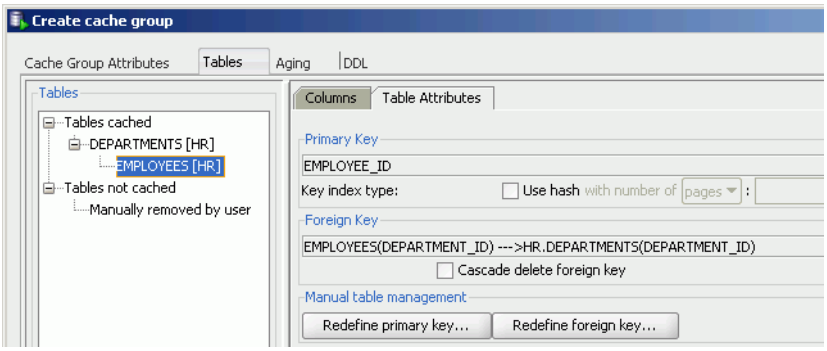
In the **Table Attributes** tab within the **Tables** tab of the **Create cache group** dialog box, specify whether to create a range index or a hash index on the child cache table's primary key columns, and define a different column or set of columns as the primary key of the cache table than the corresponding Oracle table in a similar fashion as was specified for the root table. Also, you can specify an optional WHERE clause 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.



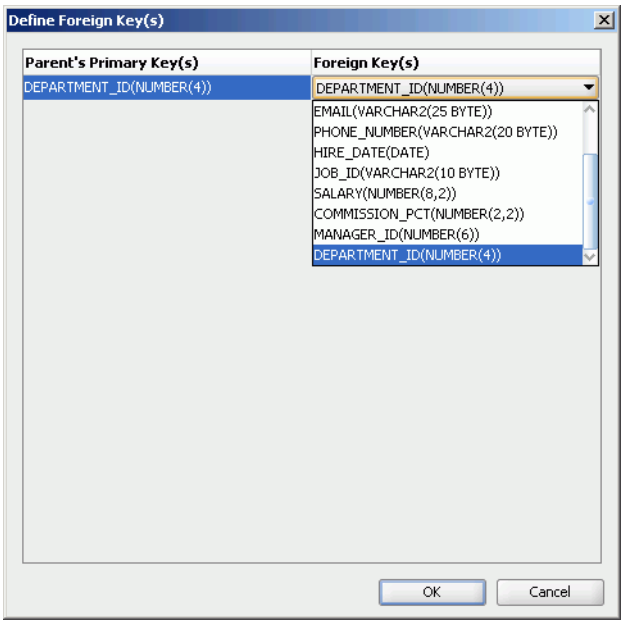
If you would like the foreign key columns of the child table to have cascade delete enabled such that when rows containing referenced key values are deleted from its parent table, rows in the child table with dependent foreign key values are also deleted, put a check in the **Cascade delete foreign key** check box.

By default, the foreign key columns of a child cache table is the same as the foreign key columns of the corresponding Oracle table. You can define a different column or set of columns as the foreign key of the cache table.

In the **Manual table management** section of the **Table Attributes** tab within the **Tables** tab of the **Create cache group** dialog box, click the **Redefine foreign key** button.

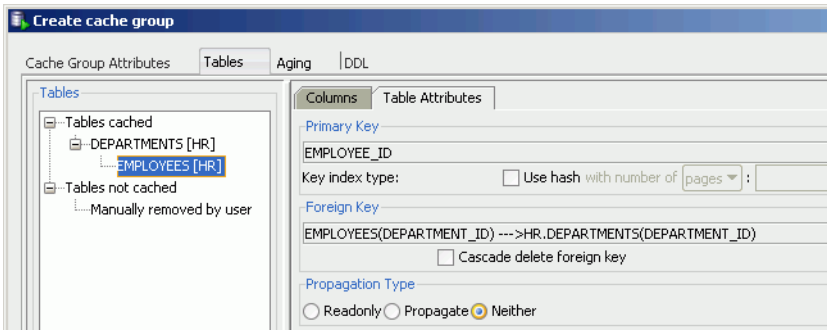


In the **Define Foreign Key(s)** dialog box, the list of columns in the child table whose data types are compatible with the primary key columns of the root table are shown in the **Foreign Key(s)** pull-down menu. Select the columns that you want the foreign key to consist of. Click the **OK** button to define the new set of foreign key columns.



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 In-Memory Database Cache User's Guide* or "Defining multiple cache group tables" in the *TimesTen Cache Connect to Oracle Guide*.

If the cache table is in a user managed cache group that has selected the **Varies from table to table** button in the **Table attributes** section within the **Cache Group Attributes** tab of the **Create cache group** dialog box, select whether the cache table is read-only or updatable in a similar fashion as was specified for the root table. If the table is updatable, specify whether updates on the table are to be or not to be propagated to its corresponding Oracle table.



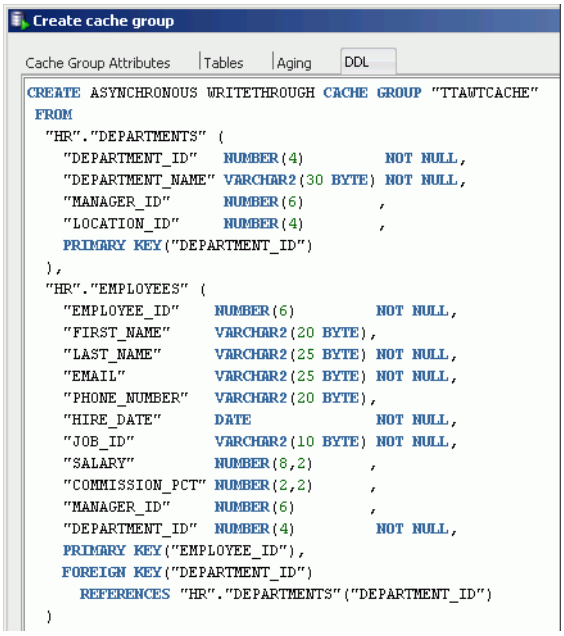
Complete the definitions of the child tables

Define the columns to cache, type of index to create on the primary key columns, cascade delete setting for the foreign key columns, and the table attributes including an optional WHERE clause for any remaining child tables.

Verify the cache group definition

In the **DDL** tab of the **Create cache group** dialog box, you can view the CREATE CACHE GROUP statement that will be used to create the cache group and its cache tables.

Click the **Apply** button to create the cache group and its cache tables.



If you created an AWT cache group, you must start the replication agent in order for committed updates on its cache tables to be asynchronously propagated to its corresponding Oracle tables. See "Start the replication agent" on page 6-4 for information on how to start the replication agent.

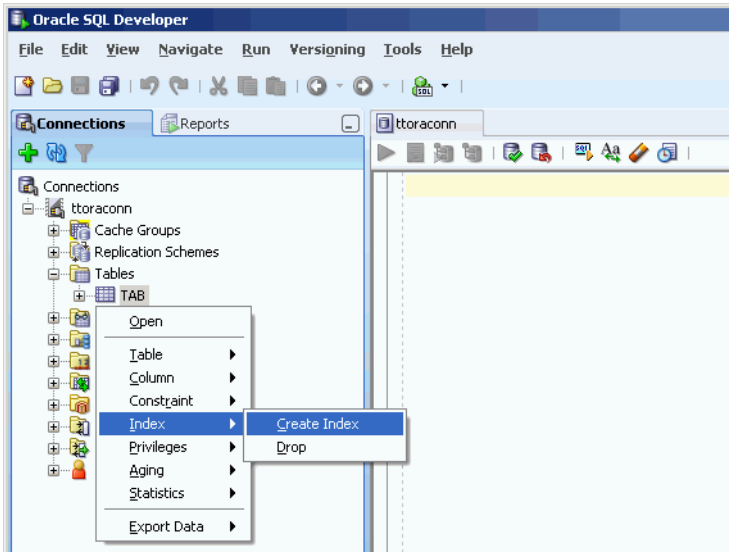
If you created a global cache group, you must attach the TimesTen database to the cache grid in order for data in its cache tables to maintain consistency among the TimesTen databases within the grid. This task must be performed outside of SQL Developer using a command-line interface such as the ttIsql utility, or within SQL Developer using SQL Worksheet. See "Attaching a TimesTen database to a cache grid" in the *Oracle In-Memory Database Cache User's Guide* for information.

Note: Global cache groups and cache grids are supported in TimesTen 11*g* and later.

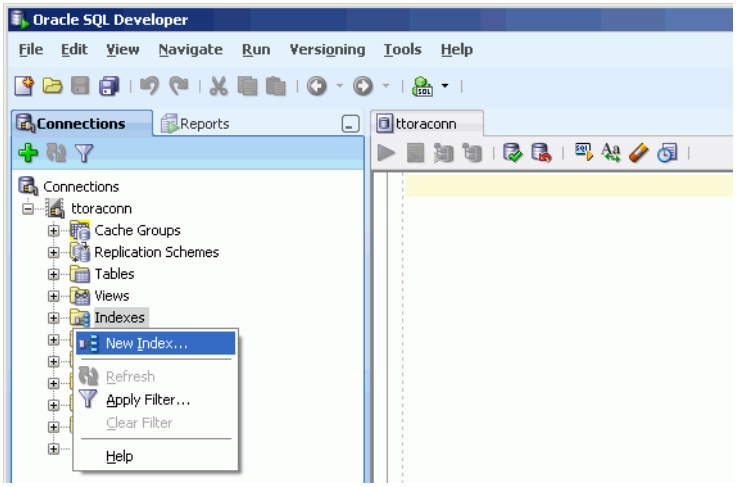
Create an index on a table

You can create a range index or a bitmap 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 that you want to create an index on and select **Index** > **Create Index**.



You can also right-click the **Indexes** node and select **New Index**.



In the **Properties** tab of the **Create Index** dialog box, select the table that you want to create an index on from the **Table** pull-down menu.

In the **Type** section, select the:

- **Non-unique** button to create a non-unique range index,
- **Unique** button to create a unique range index, or
- **Bitmap** button to create a bitmap index

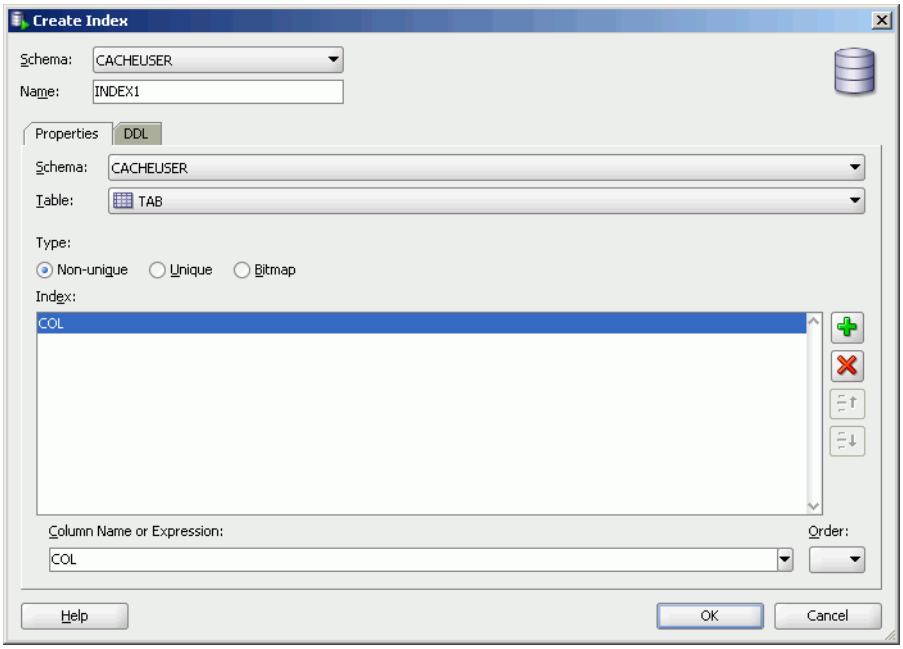
See "CREATE INDEX" in the *Oracle TimesTen In-Memory Database SQL Reference* for information about the different types of indexes supported in TimesTen.

Note: Bitmap indexes are supported in TimesTen 11*g* and later.

In TimesTen 7.0, a range index is called a T-tree index.

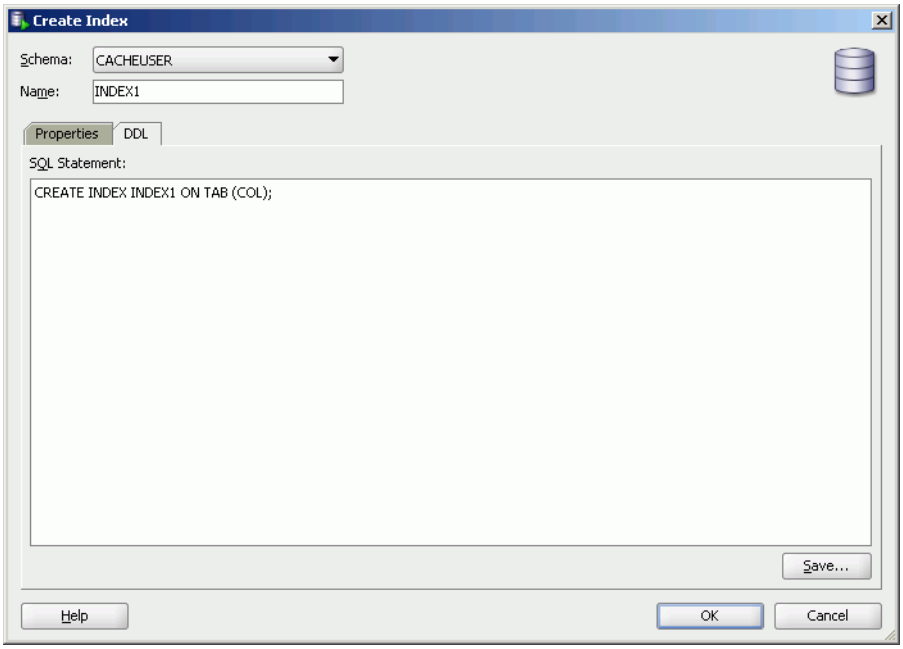
In the **Index** section, specify the columns that the index is to be defined on. The columns can be selected from the **Column Name or Expression** pull-down menu. For each indexed column, you can specify whether the column is to be sorted in ascending or descending order from the **Order** pull-down menu. The default sort order is ascending.

To create a composite index, click the **+** button to add columns to the index. Click the **X** button to remove columns from the index.



In the **DDL** tab of the **Create Index** dialog box, you can view the CREATE INDEX statement that will be used to create the index.

Click the **OK** button to create the index.

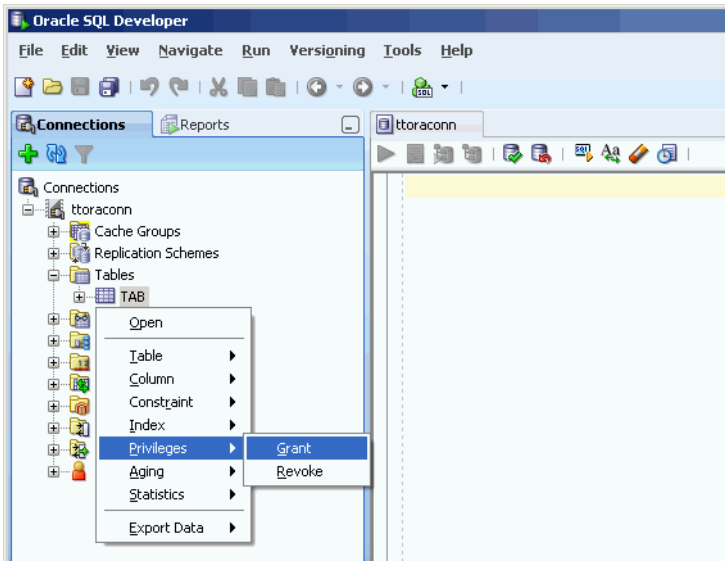


Grant and revoke object privileges to and from database elements

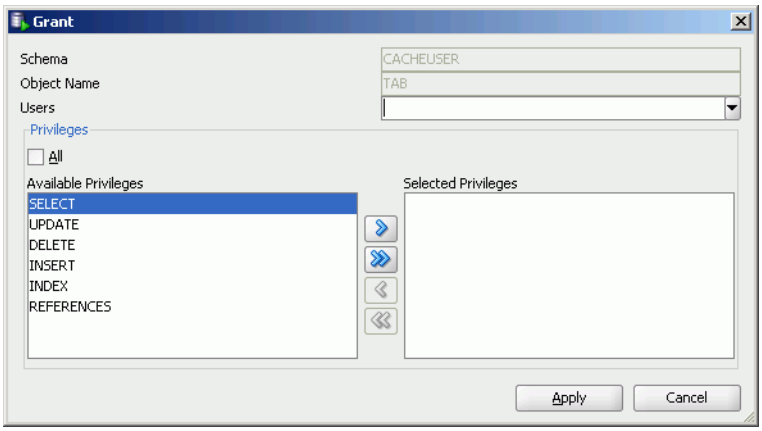
You can grant privileges to and revoke privileges from the following database elements:

- Regular and cache tables
- Regular and materialized views
- Sequences
- PL/SQL packages, procedures and functions (applies only to TimesTen 11*g* and later)

To grant privileges on a particular element, right-click the name of the element and select **Privileges** > **Grant**.



In the **Users** pull-down menu of the **Grant** dialog box, select the user that you want to grant object privileges to that element. In the **Privileges** section, put a check in the **All** check box or click the **>>** button if you want to grant all available object privileges on the element to the selected user. Otherwise, select the individual privileges from the **Available Privileges** list that you want to grant to the selected user and then click the **>** button to move those privileges into the **Selected Privileges** list. To select more than one privilege, press and hold the CTRL key, and click the desired privileges. Click the **Apply** button to grant the selected object privileges on the element to the selected user.

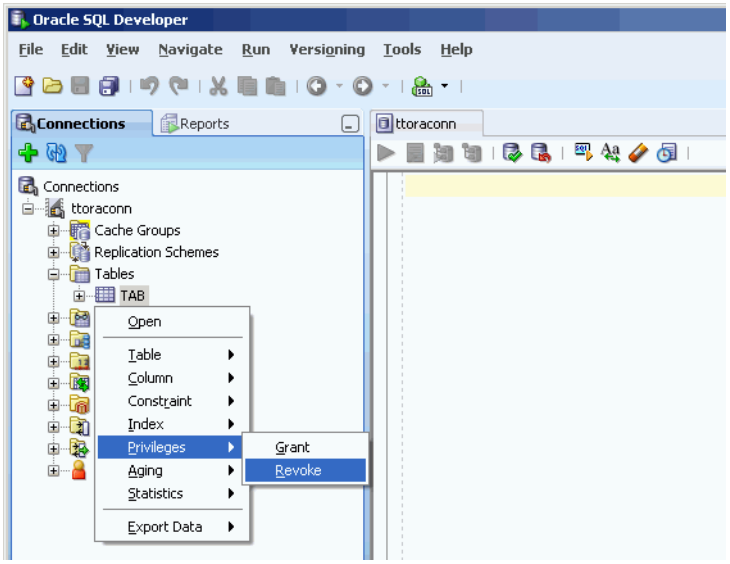


For a particular element type, only the available object privileges are shown. For example, DELETE, INDEX, INSERT, REFERENCES, SELECT and UPDATE privileges can be granted on a table. However, only INDEX, REFERENCES and SELECT privileges can be granted on a materialized view.

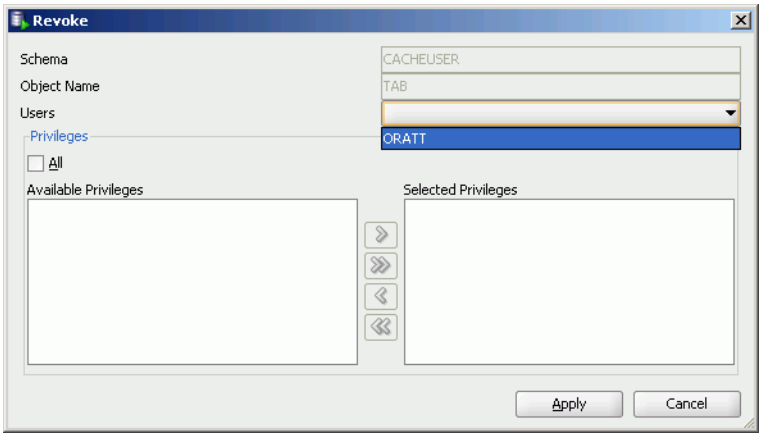
Only users with the ADMIN system privilege or the owner of the particular element can grant object privileges on that element.

See "Object privileges" in the *Oracle TimesTen In-Memory Database SQL Reference* for more information on the set of privileges available to each element type.

To revoke privileges from a particular element, right-click the name of the element and select **Privileges** > **Revoke**.



In the **Users** pull-down menu of the **Revoke** dialog box, select the user that you want to revoke object privileges from that element. In the **Privileges** section, put a check in the **All** check box or click the **>>** button if you want to revoke all granted object privileges on the element from the selected user. Otherwise, select the individual privileges from the **Available Privileges** list that you want to revoke from the selected user and then click the **>** button to move those privileges into the **Selected Privileges** list. To select more than one privilege, press and hold the CTRL key, and click the desired privileges. Click the **Apply** button to revoke the selected object privileges on the element from the selected user.



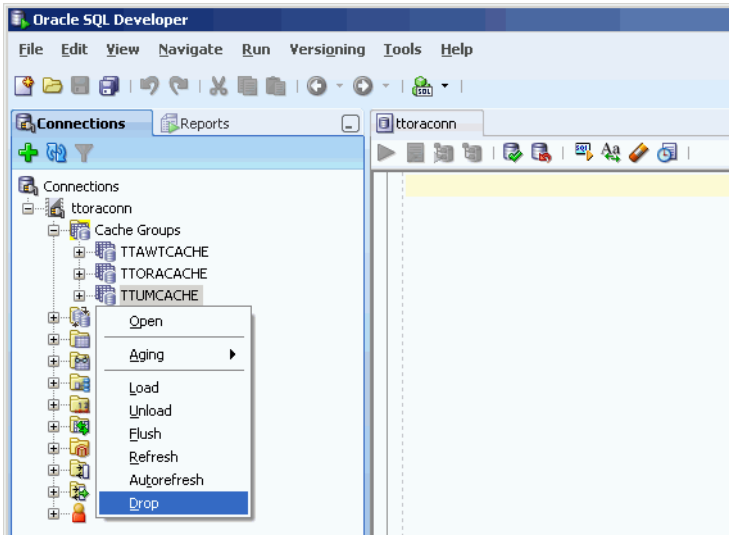
The **Users** pull-down menu only shows users who have been granted object privileges on the element. For a particular user, only the object privileges that they have been granted are shown.

Only users with the ADMIN system privilege or the owner of the particular element can revoke object privileges from that element.

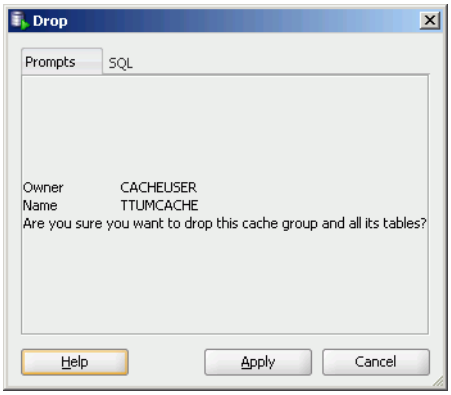
Drop a cache group

When you drop a cache group, its corresponding cache tables are also dropped.

Under the **Cache Groups** node within the node that contains the name of your database connection, right-click the name of the cache group that you want to drop and select **Drop**.



In the **Prompts** tab of the **Drop** dialog box, verify that you want to drop the cache group. Then click the **Apply** button to drop the cache group.



For more information about dropping a cache group, see "Dropping a cache group" in the *Oracle In-Memory Database Cache User's Guide* or the *TimesTen Cache Connect to Oracle Guide*.

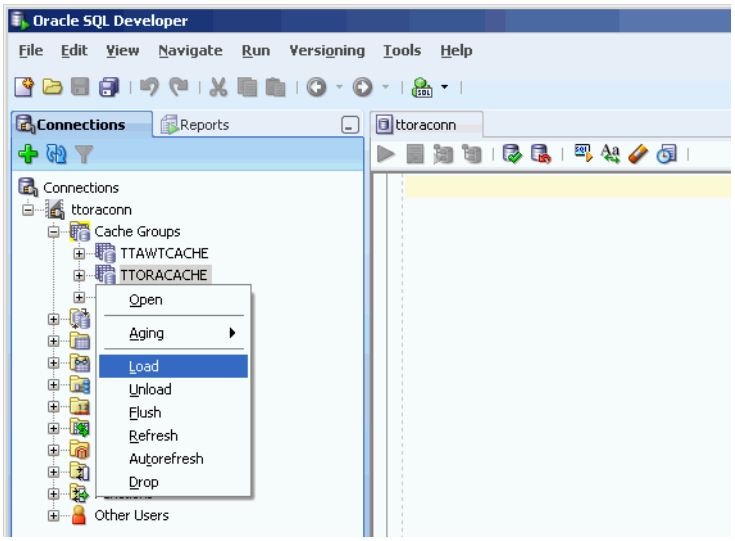
4

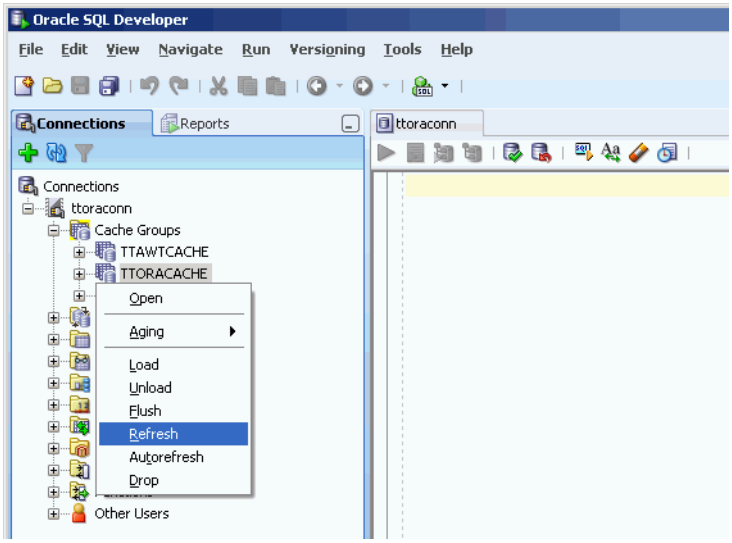
Cache Group Operations

After you have created a cache group, you can perform operations to load, refresh, unload or flush the cache group depending on the cache group type.

Load or refresh 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 that you want to load or refresh. Then select **Load** to load the cache group or **Refresh** to refresh the cache group.

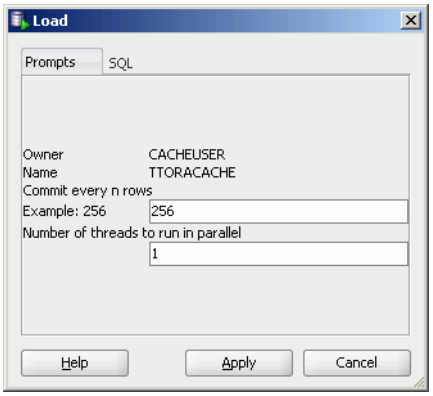


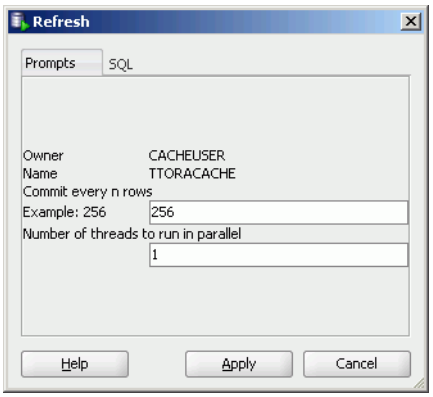


The cache agent must be running in order to load or refresh a cache group. See "Start and stop the cache agent" on page 2-3 for information about how to start the cache agent.

For a read-only cache group or a user managed cache group that has automatic refresh defined, its automatic refresh state must be set to Paused before you can load or refresh the cache group. For information about changing the automatic refresh state of a cache group, see "Alter the automatic refresh attributes of a cache group" on page 5-10.

In the **Prompts** tab of the **Load** or **Refresh** dialog box, 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 will be processed serially. For more information about parallel loading or refreshing of cache groups, see "Improving the performance of loading and refreshing a large number of cache instances" in the *Oracle In-Memory Database Cache User's Guide* or "Improving performance of loading and refreshing large tables" in the *TimesTen Cache Connect to Oracle Guide*.

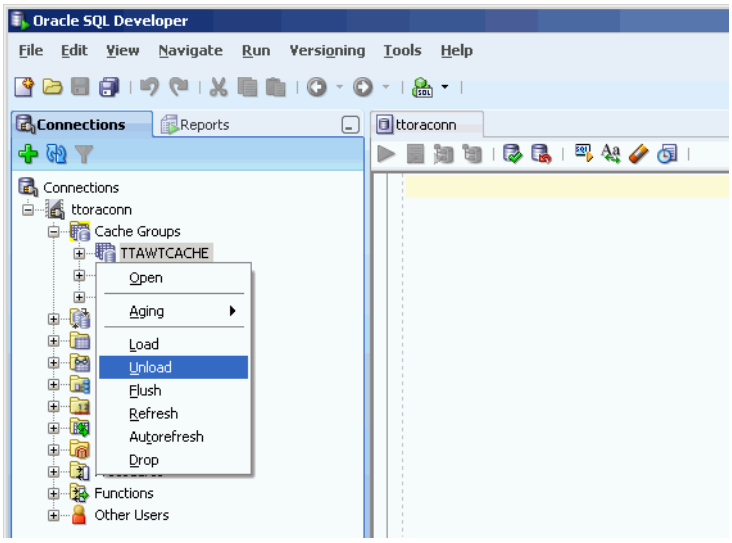
You can specify an optional WHERE clause for the load or refresh operation in the **Where clause** field within the **Prompts** tab of the **Load** or **Refresh** dialog box. A WHERE clause is only allowed for a load or refresh operation on an AWT, SWT or user managed cache group that does not have automatic 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 In-Memory Database Cache User's Guide* or the *TimesTen Cache Connect to Oracle Guide*.

Click the **Apply** button 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 In-Memory Database Cache User's Guide* or the *TimesTen Cache Connect to Oracle Guide*.

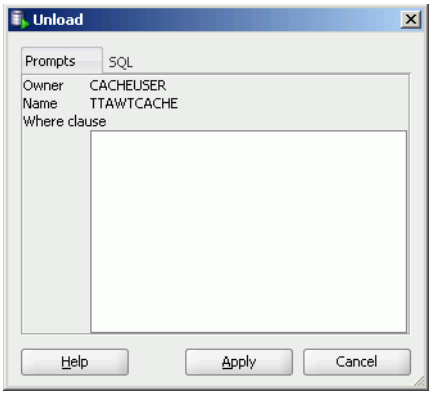
Unload a cache group

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 that you want to unload. Then select **Unload** to unload the cache group.



In the **Prompts** tab of the **Unload** dialog box, you can specify an optional WHERE clause for the unload operation in the **Where clause** field. A WHERE clause is only allowed for an unload operation on an AWT, SWT or user managed cache group that does not have automatic refresh defined.

Click the **Apply** button to unload the cache group.

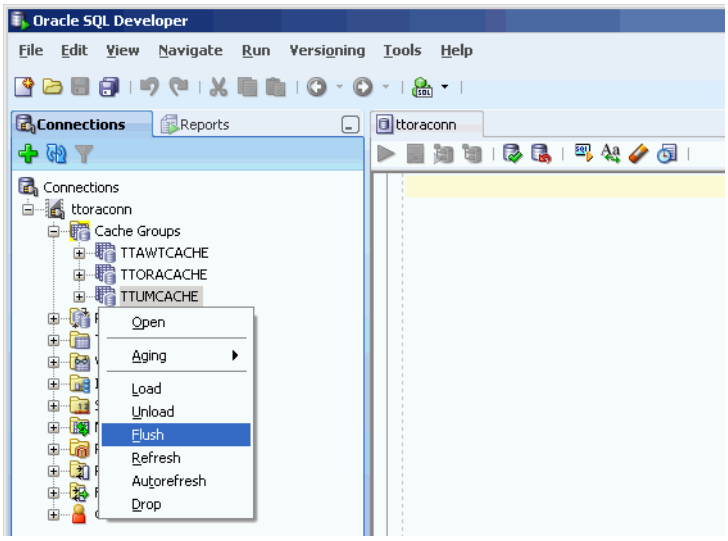


For more information about unloading a cache group, see "Unloading a cache group" in the *Oracle In-Memory Database Cache User's Guide* or the *TimesTen Cache Connect to Oracle Guide*.

Flush a cache group

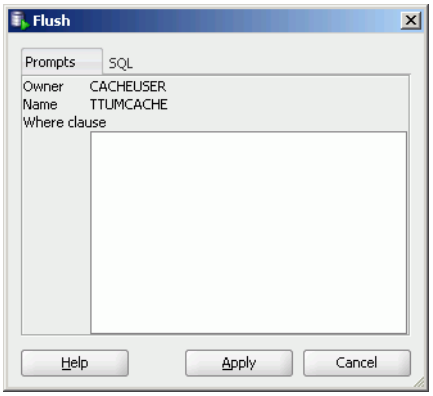
Only user managed cache groups containing cache tables that are not all read-only, or not all updatable where updates on the tables are to be propagated to its corresponding Oracle tables can be flushed.

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 want to flush. Then select **Flush** to flush the cache group.



In the **Prompts** tab of the **Flush** dialog box, you can specify an optional WHERE clause for the flush operation in the **Where clause** field.

Click the **Apply** button to flush the cache group.



For more information about flushing a cache group, see "Flushing a user managed cache group" in the *Oracle In-Memory Database Cache User's Guide* or "Flushing a USERMANAGED cache group" in the *TimesTen Cache Connect to Oracle Guide*.

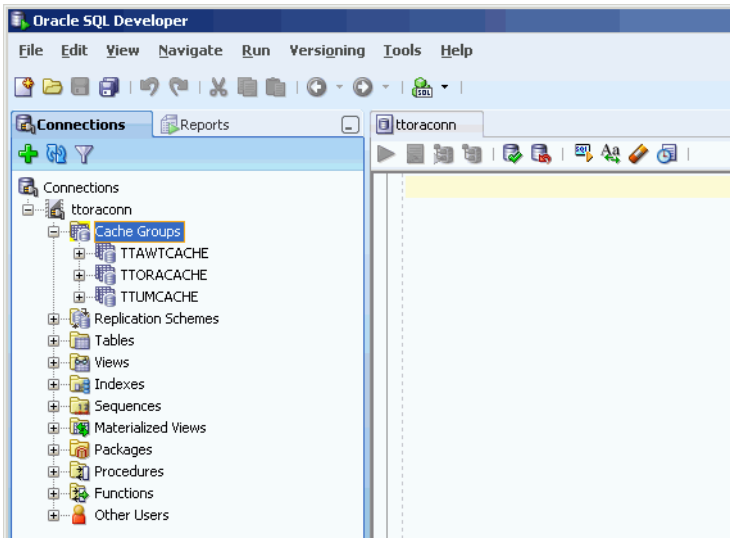
5

Cache Group and Cache Table Characteristics

After you have created a cache group, you can view the characteristics of the cache group itself such as its type and automatic refresh attributes, and its cache tables. You can also alter the cache group's automatic refresh attributes and aging policy.

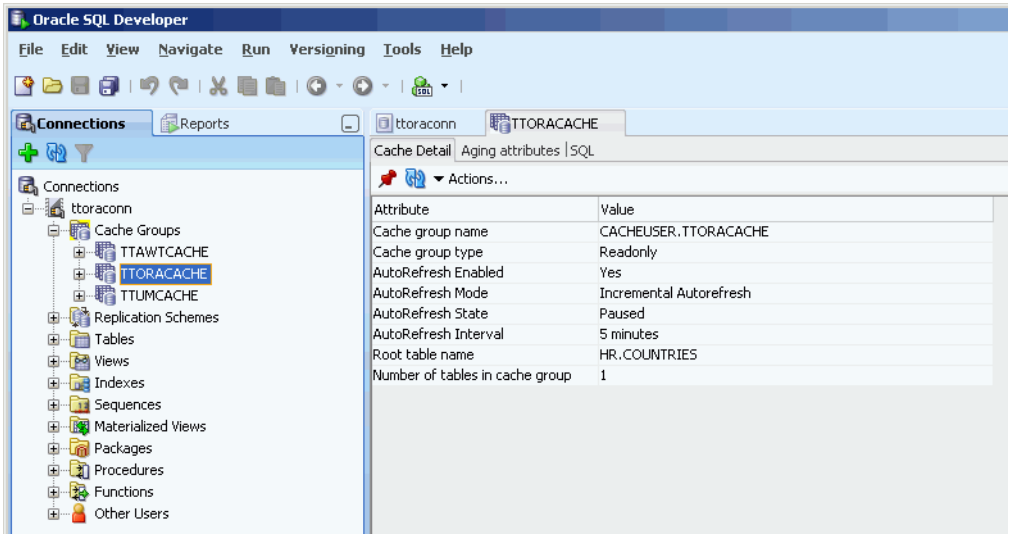
View the characteristics of a cache group

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 that you want to view the characteristics of.



In the **Cache Detail** tab within the cache group's tab, it 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, it also shows the automatic refresh mode, interval and state.

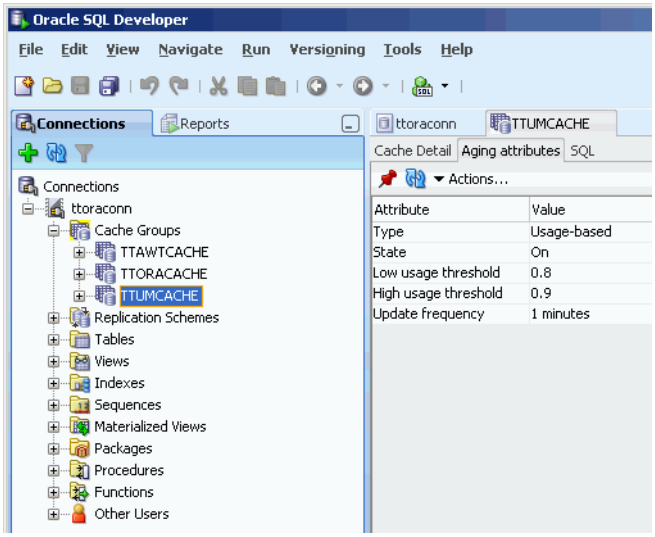


View the aging attributes

In the **Aging attributes** tab within the cache group's tab, it shows information such as the aging policy type, aging cycle, and aging state for cache groups that have an aging policy defined.

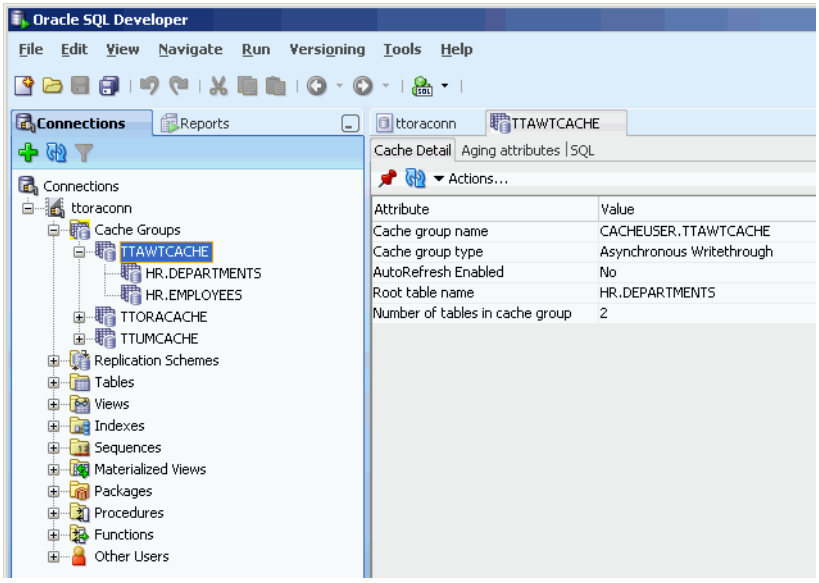
For cache groups that have an LRU aging policy defined, it also shows the TimesTen database memory usage thresholds that determine when data starts and stops being removed from the cache tables, and the LRU aging cycle. See "Specify an aging policy" on page 3-8 for information about how to change these thresholds and the LRU aging cycle.

For cache groups that have a time-based aging policy defined, it also shows 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 in which data that has not been updated is to be kept in the table.



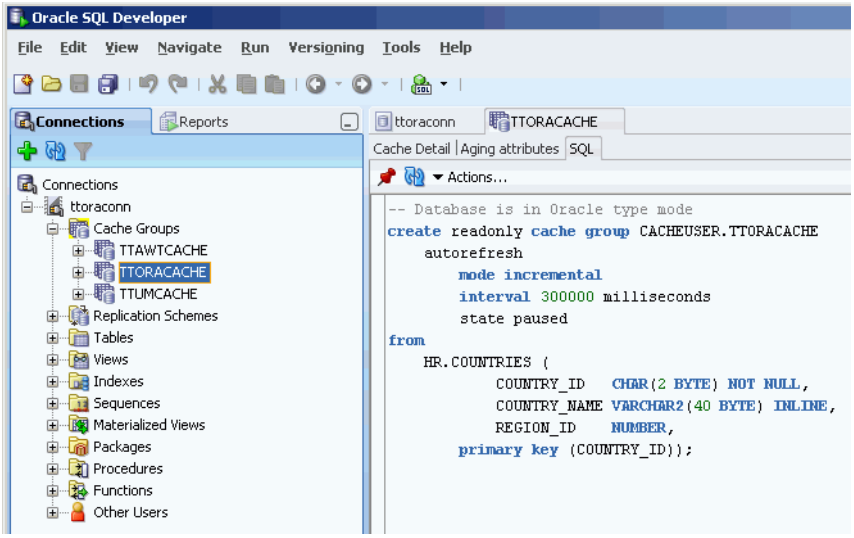
View the cache tables

To view the list of cache tables in a cache group, under the **Cache Groups** node, click the **+** to the left of the node that contains the name of the cache group.



View the cache group definition

In the **SQL** tab within the cache group's tab, you can view the CREATE CACHE GROUP statement that was used to create the cache group.

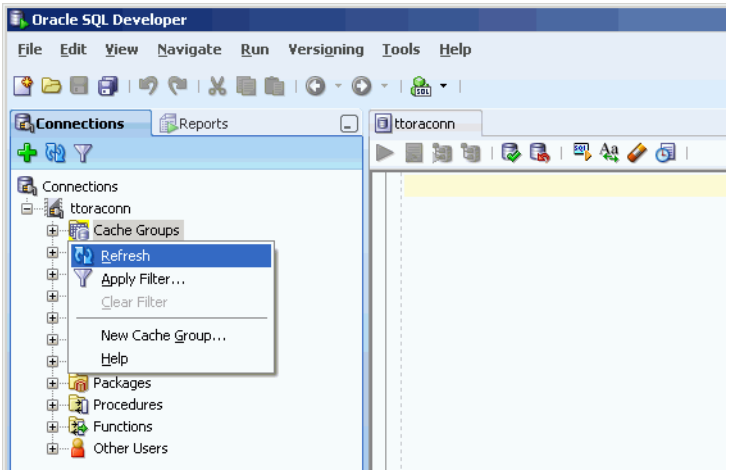


If you are unable to find the cache group that you are looking for, see "Locate a TimesTen database element" on page 5-4 for information.

Locate a TimesTen database element

If you are unable to find the database element that you are looking for, right-click the node corresponding to the element type and select **Refresh** to refresh the list of elements of that type.

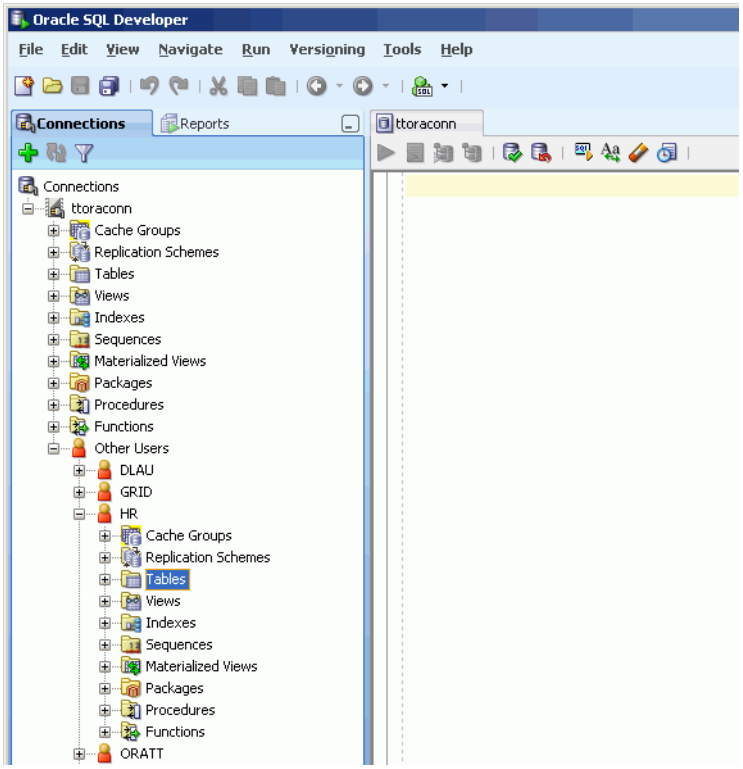
For example, to find a cache group, right-click the **Cache Groups** node and select **Refresh** to refresh the list of cache groups.



You can also check whether the element 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 element type to view the list of elements of that type owned by that user.

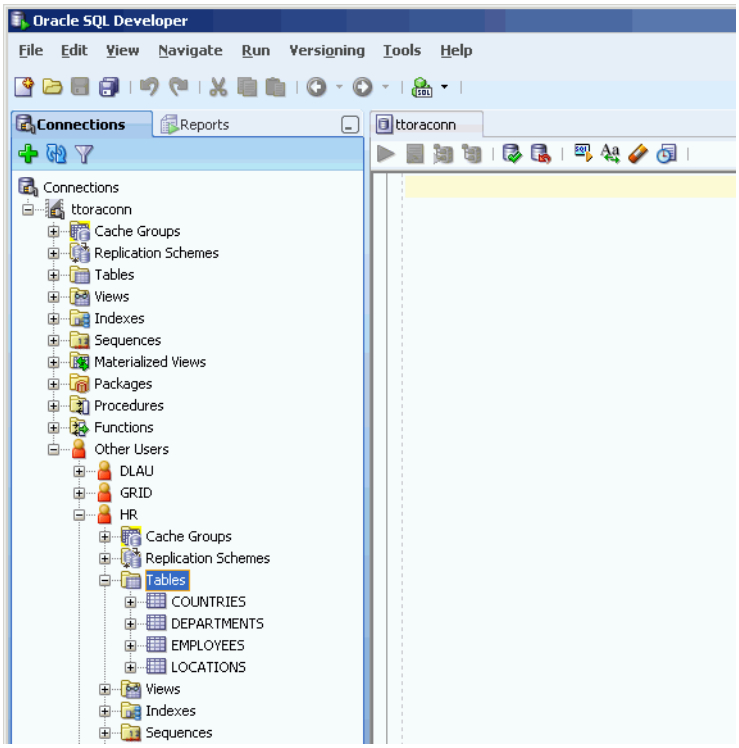
For example, to find a table, click the **+** to the left of the user's **Tables** node to view the list of tables owned by that user. Click the **+** to the left of the node that contains the name of a table to view the list of columns in the table.



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 that you want to view the characteristics of.

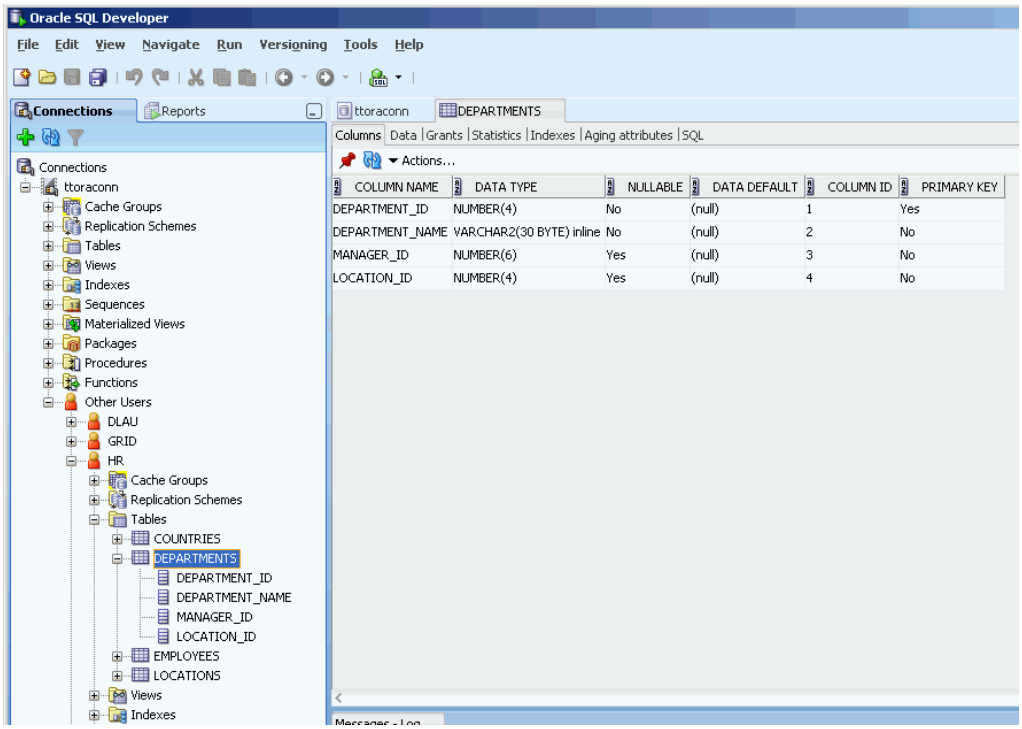


If you are unable to find the table that you are looking for, see "Locate a TimesTen database element" on page 5-4 for information.

View the columns

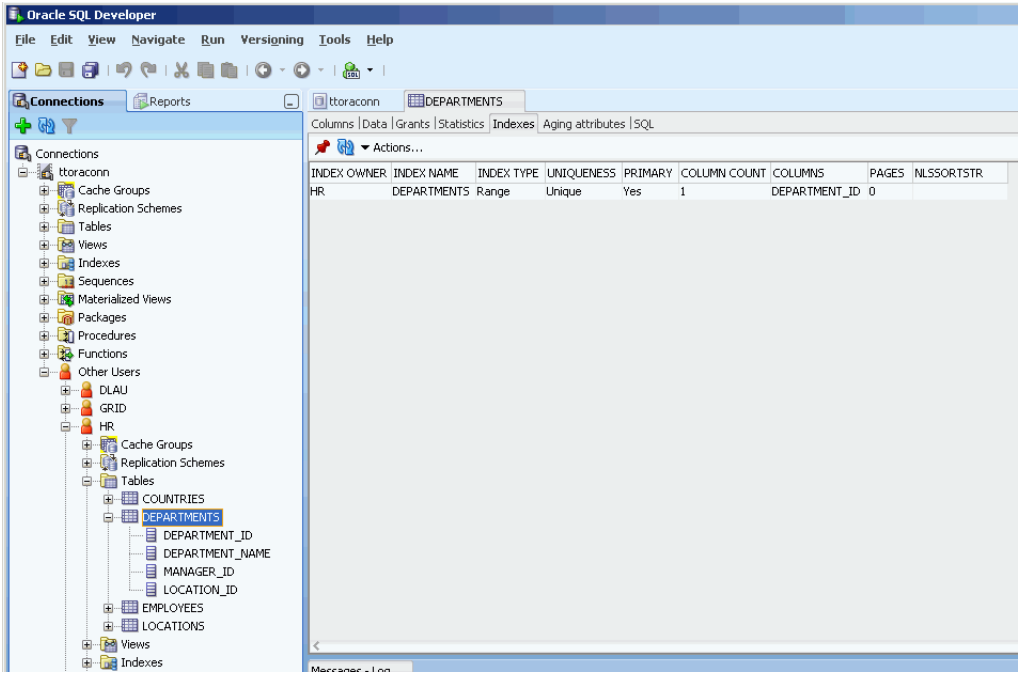
In the **Columns** tab within the table's tab, it shows information such as the name, data type, nullability property and default value of the table's columns.

Under the **Tables** node, you can also view the list of columns in a table by clicking the + to the left of the node that contains the name of the table.



View the indexes

In the **Indexes** tab within the table's tab, it shows information such as the name, type, uniqueness property, and the number and name of the columns of the table's indexes.

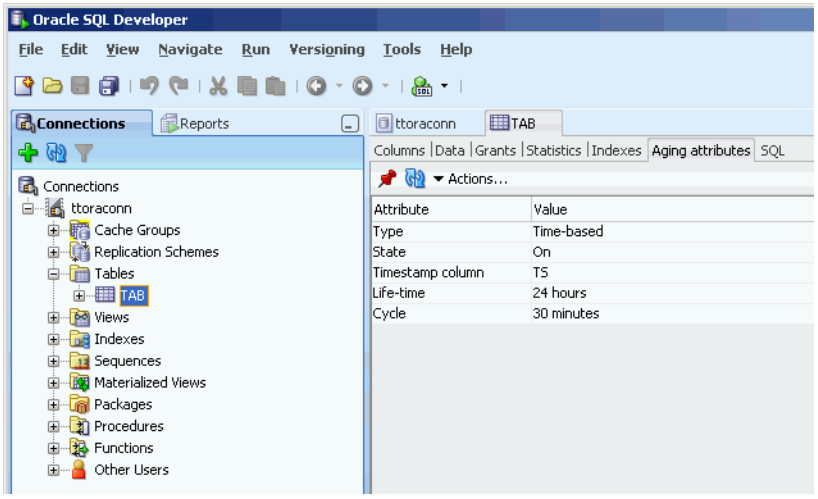


View the aging attributes

In the **Aging attributes** tab within the table's tab, it shows information such as the aging policy type, aging cycle, and aging state for tables that have an aging policy defined.

For tables that have an LRU aging policy defined, it also shows the TimesTen database memory usage thresholds that determine when data starts and stops being removed from the tables, and the LRU aging cycle. See "Specify an aging policy" on page 3-8 for information about how to change these thresholds and the LRU aging cycle.

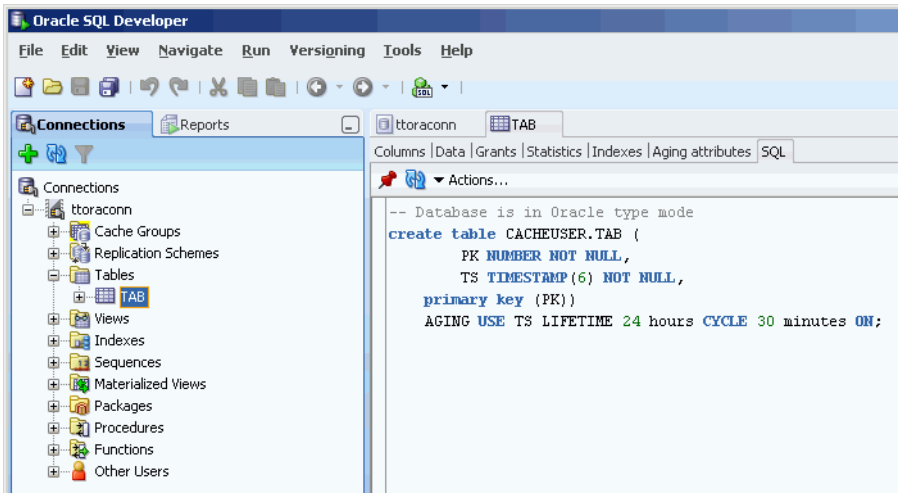
For tables that have a time-based aging policy defined, it also shows the name of the non-nullable DATE or TIMESTAMP column used to store the timestamp value indicating when each row was added or most recently updated, and the length of time in which data that has not been updated is to be kept in the table.



View the table definition

In the **SQL** tab within the table's tab, you can view the CREATE TABLE statement that was used to create the table for a regular 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 "View the cache group definition" on page 5-4 for more information.

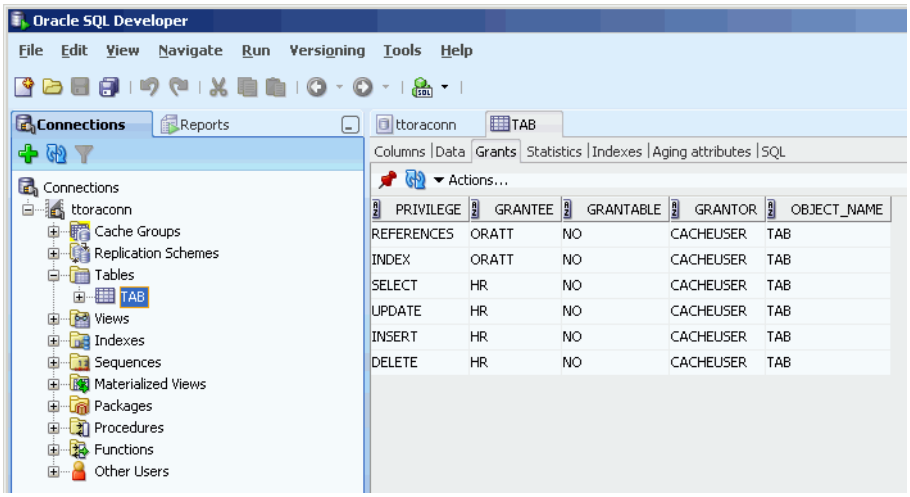


View the privileges granted on a database element

You can view the object privileges granted to all users on a particular element 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 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.

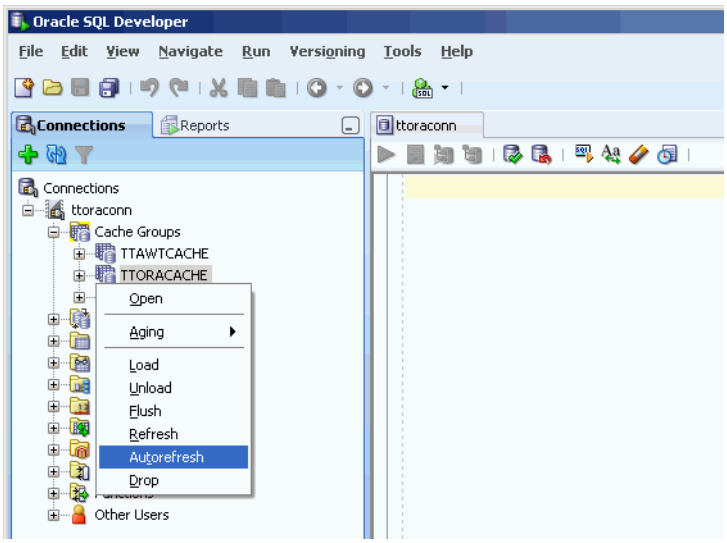
In the **Grants** tab within the table's tab, it shows the users who have been granted privileges on the table, what privileges they have been granted, and which user granted the privileges.



Alter 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.

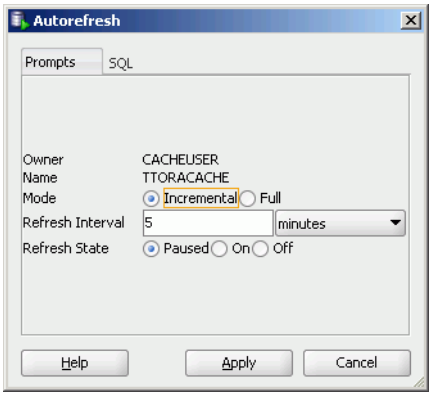
Under the **Cache Groups** node, right-click the name of the cache group that you want to alter the automatic refresh attributes and select **Autorefresh**.



In the **Prompts** tab of the **Autorefresh** dialog box, you can change any of the following automatic refresh attributes:

- Change the automatic refresh mode by selecting either the **Incremental** or **Full** button in the **Mode** field.
- Change the automatic refresh interval by specifying a numeric interval and units of time in minutes, seconds or milliseconds from the pull-down menu in the **Refresh Interval** field.
- Change the automatic refresh state by selecting either the **Paused**, **On** or **Off** button in the **Refresh State** field.

Click the **Apply** button to alter the automatic refresh attributes of the cache group.



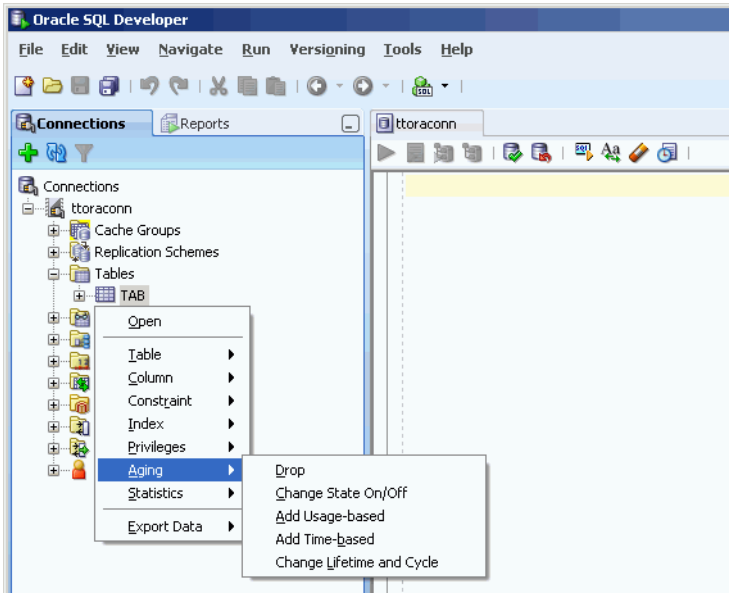
For more information about automatic refresh of cache groups, see "AUTOREFRESH cache group attribute" in the *Oracle In-Memory Database Cache User's Guid*e.

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.

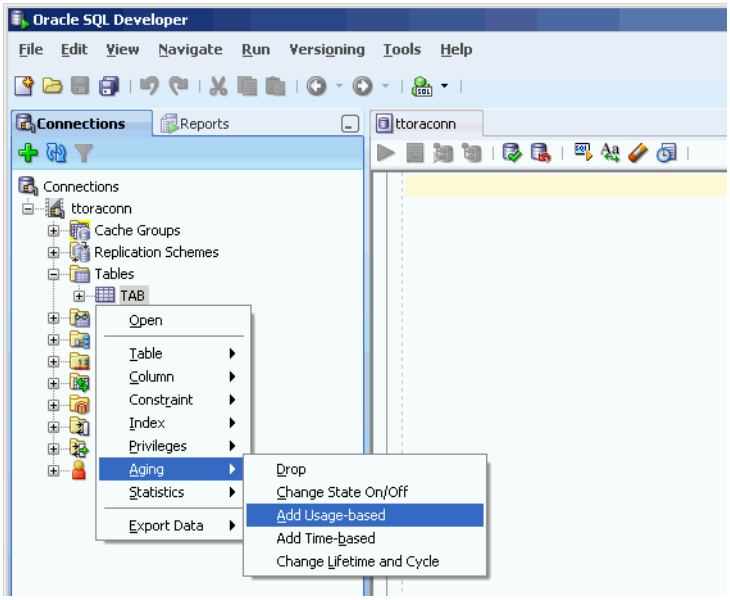
Add an aging policy to a table

To add an aging policy to a regular table or a cache table, under the **Tables** node, right-click the name of the table that you want to add an aging policy to. An aging policy can be added to a cache table only if it is the root table of a cache group.

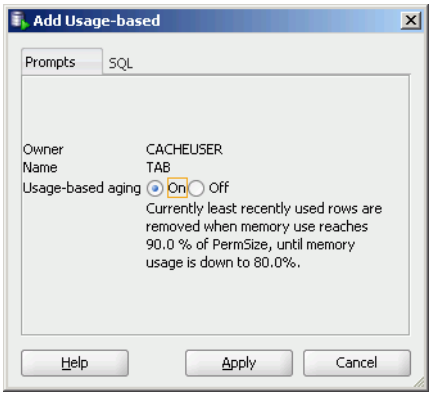


If you are unable to find the table that you are looking for, see "Locate a TimesTen database element" on page 5-4 for information.

After you right-click the name of the table that you want to add an aging policy to, select **Aging > Add Usage-based** to add an LRU aging policy or select **Aging > Add Time-based** to add a time-based aging policy.



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 selecting either the **on** or **off** button in the **Usage-based aging** field. The default aging state is on. Then click the **Apply** button to add the LRU aging policy to the table.



The TimesTen database memory usage thresholds determine when data starts and stops being removed from the table. The default memory usage threshold that determines when data starts being removed from the table is 90%. The default memory usage threshold that determines when data stops being removed from the table is 80%. The default LRU aging cycle is 1 minute. See "Specify an aging policy" on page 3-8 for information about how to change these thresholds and the LRU aging cycle.

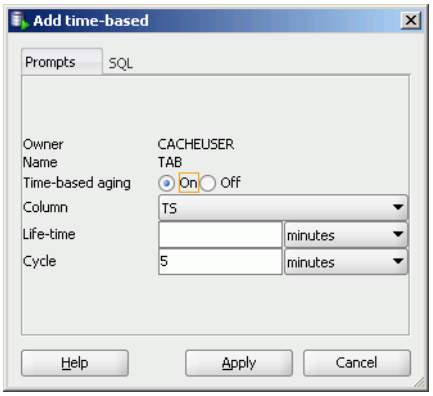
An LRU aging policy can be added to any regular table, and only to cache tables in an AWT, SWT or user-managed cache group that does not have automatic refresh defined.

To add a time-based aging policy to the table, in the **Prompts** tab of the **Add time-based** dialog box, select the column from the **Column** pull-down menu that you want to store the timestamp value indicating when each row was added or most recently updated.

Indicate the length of time in which data that has not been updated is to be kept in the table by specifying, in the **Life-time** field, a numeric value followed by a unit of minutes, hours or days in the pull-down menu.

Indicate the frequency at which data is to be aged out of the table by specifying, in the **Cycle** field, a numeric value followed by a unit of minutes, hours or days in the pull-down menu. The default time-based aging cycle is 5 minutes.

Specify an aging state by selecting either the **on** or **off** button in the **Time-based aging** field. The default aging state is on. Then click the **Apply** button to add the time-based aging policy to the table.



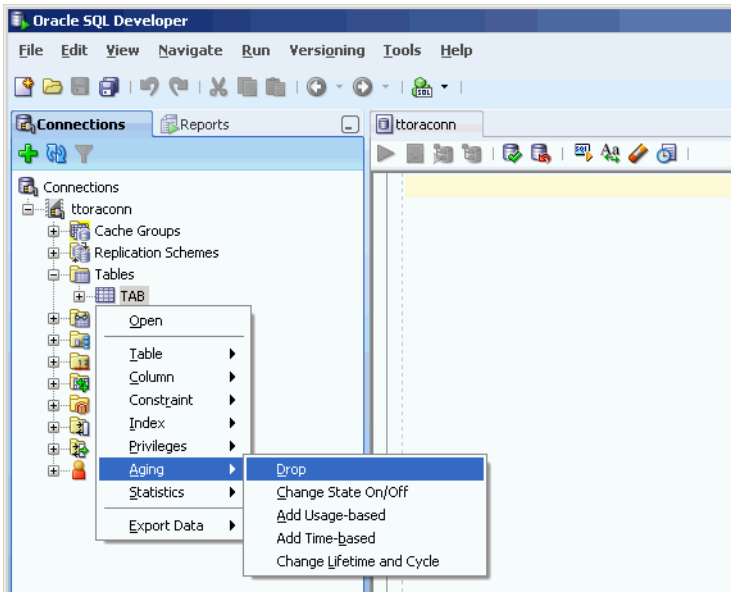
A time-based aging policy can only be added to a table that contains a non-nullable DATE or TIMESTAMP column.

Drop 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 that you want to drop an aging policy from.

If you are unable to find the table that you are looking for, see "Locate a TimesTen database element" on page 5-4 for information.

After you right-click the name of the table that you want to drop an aging policy from, select **Aging > Drop**.



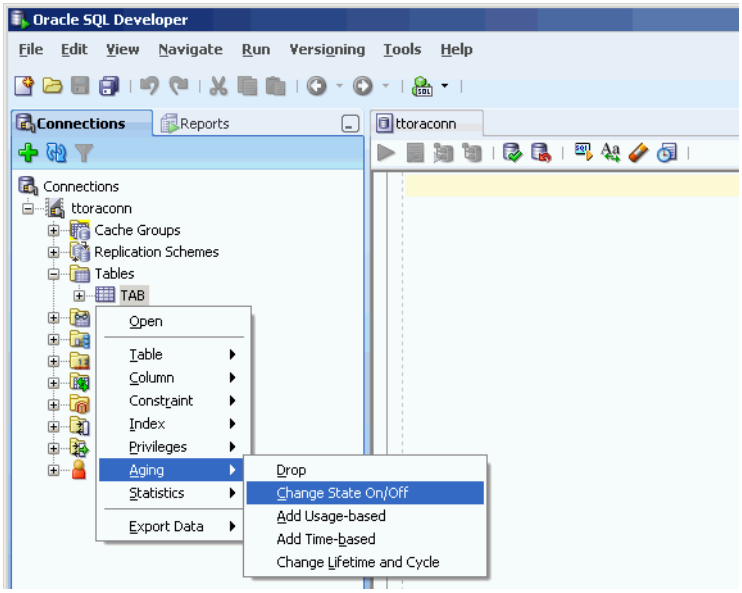
In the **Prompts** tab of the **Drop** dialog box, verify that you want to drop the aging policy from the table. Then click the **Apply** button to drop the aging policy from the table.

Change the aging state for a table's aging policy

To change the aging state for a regular table's or cache table's existing aging policy, under the **Tables** node, right-click the name of the table that you want to change the aging state for.

If you are unable to find the table that you are looking for, see "Locate a TimesTen database element" on page 5-4 for information.

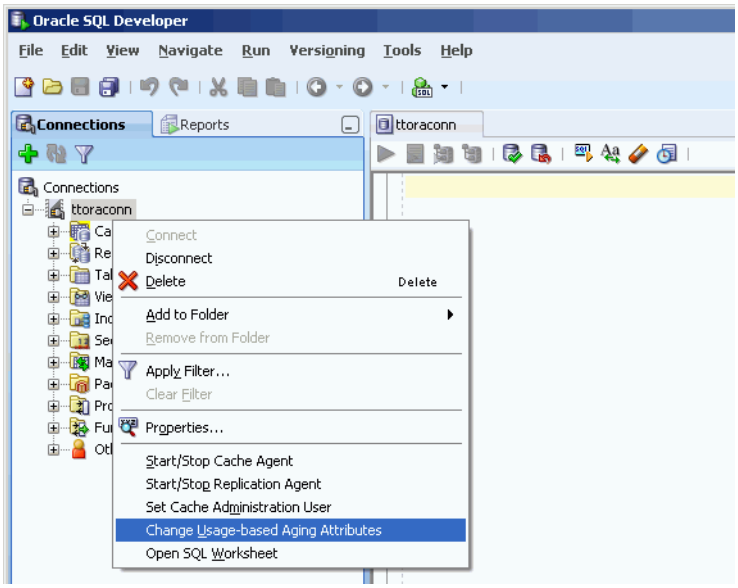
After you right-click the name of the table that you want to change the aging state for, select **Aging > Change State On/Off**.



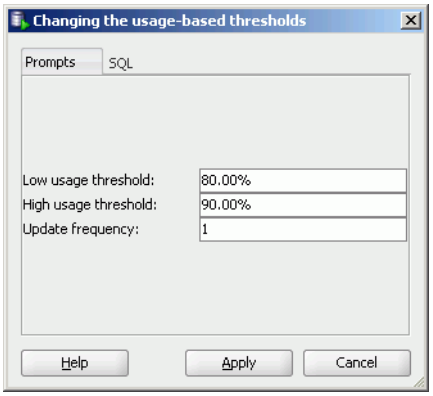
In the **Prompts** tab of the **Change state on/off** dialog box, change the aging state by selecting either the **on** or **off** button in the **Change aging state** field. Then click the **Apply** button to change the aging state for the table.

Change the memory usage thresholds and LRU aging cycle

To change the memory usage thresholds and the LRU aging cycle, right-click the node that contains the name of your database connection and select **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 removed from the tables in the **High usage threshold** field. Specify the threshold that determines when data stops being removed from the tables in the **Low usage threshold** field. Specify the LRU aging cycle in the **Update Frequency** field. Then click the **Apply** button to change the memory usage thresholds and the LRU aging cycle.



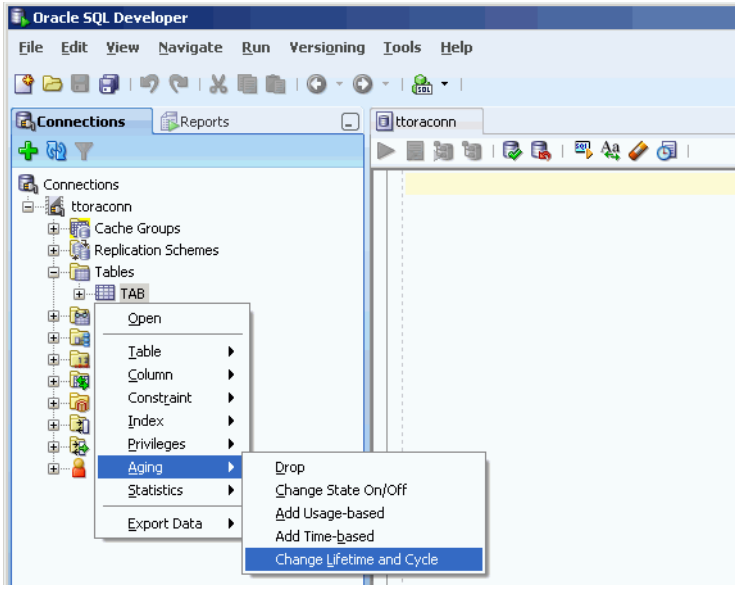
The new settings apply to all tables that have an LRU aging policy defined.

Change the lifetime and cycle for a table's time-based aging policy

To change the lifetime and cycle for a regular table's or cache table's existing time-based aging policy, under the **Tables** node, right-click the name of the table that you want to change the lifetime and cycle for.

If you are unable to find the table that you are looking for, see "Locate a TimesTen database element" on page 5-4 for information.

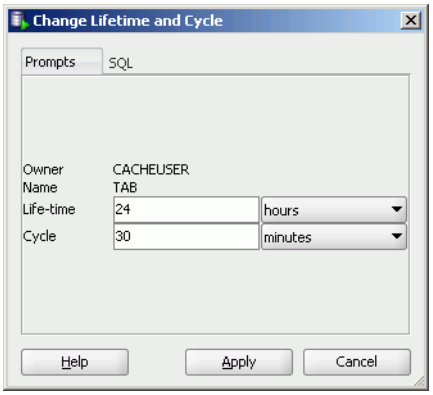
After you right-click the name of the table that you want to change the lifetime and cycle for, select **Aging > Change Lifetime and Cycle**.



In 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, in the **Life-time** field, a numeric value followed by a unit of minutes, hours or days in the pull-down menu.

Change the frequency at which data is to be aged out of the table by specifying, in the **Cycle** field, a numeric value followed by a unit of minutes, hours or days in the pull-down menu.

Click the **Apply** button to change the lifetime and cycle for the table.



6

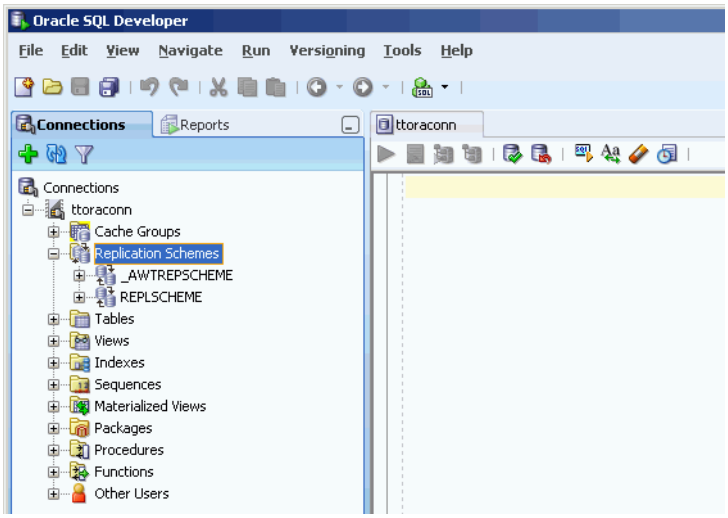
Interfacing with TimesTen Replication

A replication scheme defines the set of elements that are replicated between two or more TimesTen databases when committed updates are issued on those elements. Replication can be configured at the table or database level in a unidirectional or bidirectional fashion with asynchronous or synchronous transmit.

View the characteristics of a replication scheme

To create a replication scheme, you need to manually compose and then execute a CREATE REPLICATION or CREATE ACTIVE STANDBY PAIR statement using a command-line interface such as the ttIsql utility, or within SQL Developer using SQL Worksheet. There is no graphical interface that allows you to select elements to replicate and specify replicated database attributes. 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 elements.

Click the **+** to the left of the **Replication Schemes** node to view the list of replication schemes. Click the name of the replication scheme that you want to view the characteristics of.



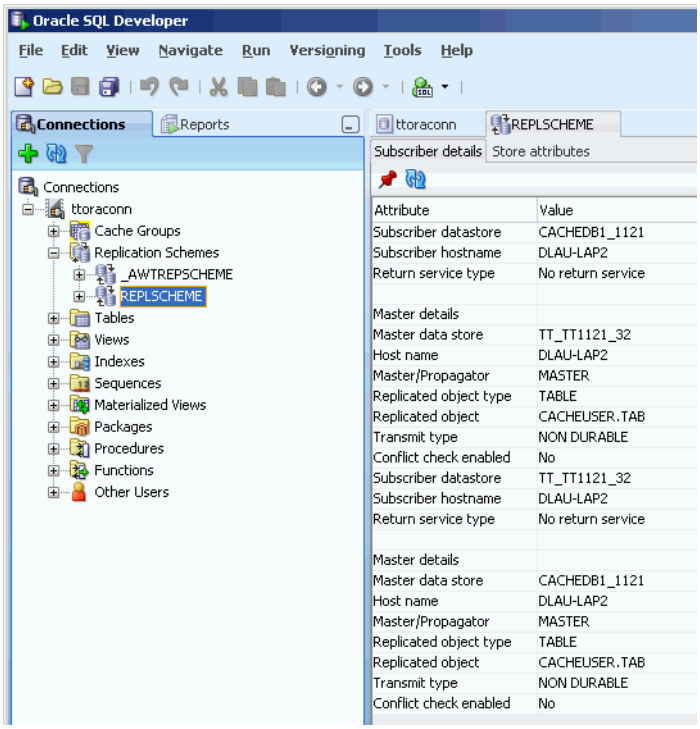
The name of the replication scheme that is automatically created to propagate updates on cache tables in AWT cache groups to the corresponding Oracle tables is _AWTREPSCHEME.

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.

If you are unable to find the replication scheme that you are looking for, see "Locate a TimesTen database element" on page 5-4 for information.

View the details of the TimesTen subscriber databases

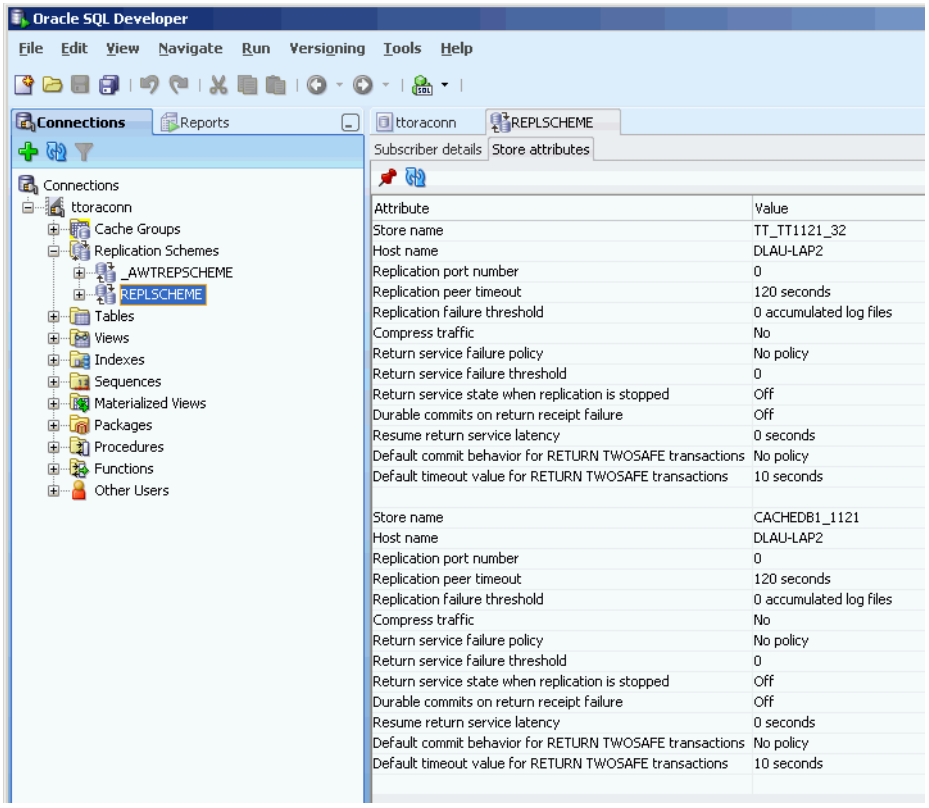
In the **Subscriber details** tab within the replication scheme's tab, it shows information such as the database name and host name of the master and subscriber TimesTen databases, the type and database name of the replicated elements, and the return service type.



View the store attributes of the replicated TimesTen databases

In the **Store attributes** tab within the replication scheme's tab, it shows information such as the TimesTen database name, host name, replication agent port number, replication response timeout period, and replication transaction log fail threshold for all databases participating in the replication scheme.

For replication schemes that have a return service defined, it also shows the return service failure policy, acknowledgement timeout period, and re-enable policy.



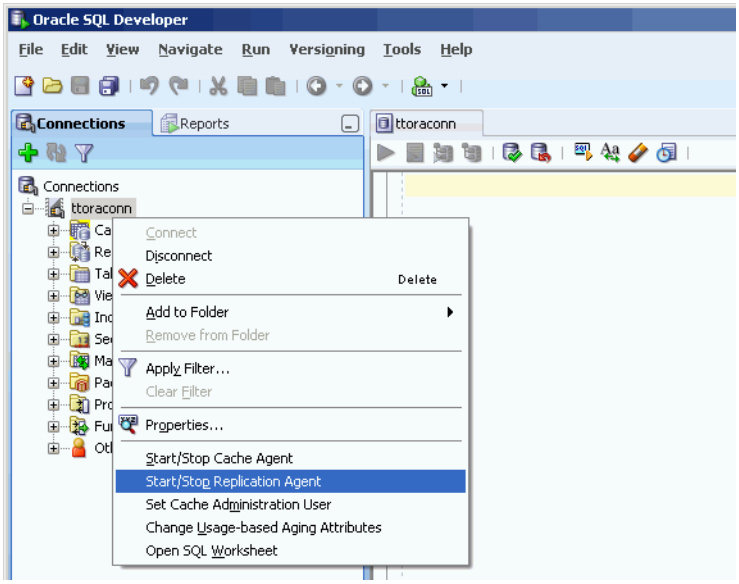
For more information about the creating a replication scheme, see "Defining Replication Schemes" in the *Oracle TimesTen In-Memory Database TimesTen to TimesTen Replication Guide* .

A replication scheme is automatically created when you create an AWT cache group. This replication scheme is owned by the TTREP user.

Start the replication agent

The replication agent process transmits committed updates on replicated tables in a master TimesTen database to replicated tables in a corresponding subscriber TimesTen database. The replication agent is also responsible for asynchronously propagating committed updates on TimesTen cache tables in AWT cache groups to its corresponding Oracle tables. After you create a replication scheme or an AWT cache group, you need to start the replication agent if it is not already running.

To start the replication agent, right-click the node that contains the name of your database connection and select **Start/Stop Replication Agent**.



In the **Starting/stopping the replication agent** dialog box, put a check in the **Start agent** check box within the **Prompts** tab. Then click the **Apply** button to start the replication agent.

The replication agent will not start if there are no replication schemes or AWT cache groups in the TimesTen database that you are connected to.

To stop a running replication agent process, right-click the node that contains the name of your database connection and select **Start/Stop Replication Agent** in an identical fashion as was done when starting the replication agent.

In the **Starting/stopping the replication agent** dialog box, put a check in the **Stop agent** check box within the **Prompts** tab. Then click the **Apply** button to stop the replication agent.

For more information about the replication agent process, see "Starting and stopping the replication agents" in the *Oracle TimesTen In-Memory Database TimesTen to TimesTen Replication Guide*, "Managing the replication agent" in the *Oracle In-Memory Database Cache User's Guide*, or "Starting the replication agent for an AWT cache group" in the *TimesTen Cache Connect to Oracle Guide*.

7

Interfacing with the TimesTen Query Optimizer

You can interact with the TimesTen query optimizer by updating the table and column statistics that are stored in the TimesTen database and used by the query optimizer to generate execution plans. You can also display the execution plan for a statement.

Update the table and column statistics used by the query optimizer

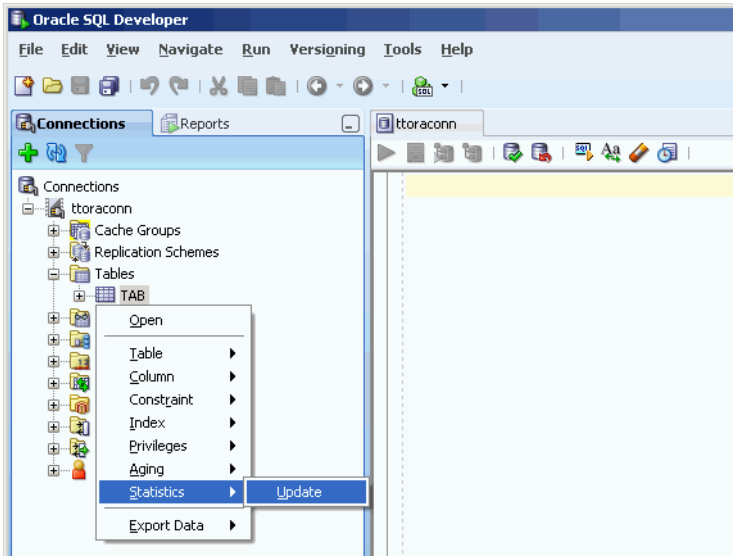
The query optimizer uses statistics stored in TimesTen database system tables to compute the optimal execution plan for a statement. Table-level statistics such as the number of rows in a table are stored in the SYS.TBL_STATS table. Column-level statistics such as the minimum and maximum value, and the number of unique values in a column are stored in the SYS.COL_STATS table.

Table and column statistics are not computed as updates occur on regular and cache tables. Instead statistics are updated when an explicit request is made to compute them.

Click the **+** to the left of the **Tables** node to view the list of tables. Right-click the name of the table that you want to update the optimizer statistics for.

If you are unable to find the table that you are looking for, see "Locate a TimesTen database element" on page 5-4 for information.

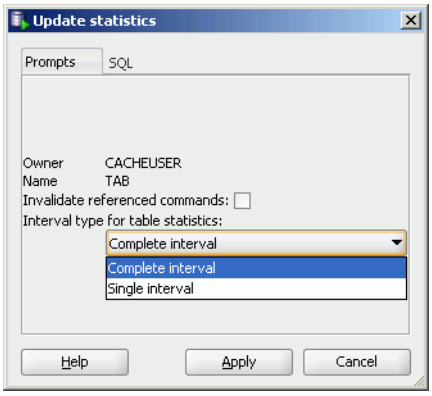
After you right-click the name of the table that you want to update the optimizer statistics for, select **Statistics > Update**.



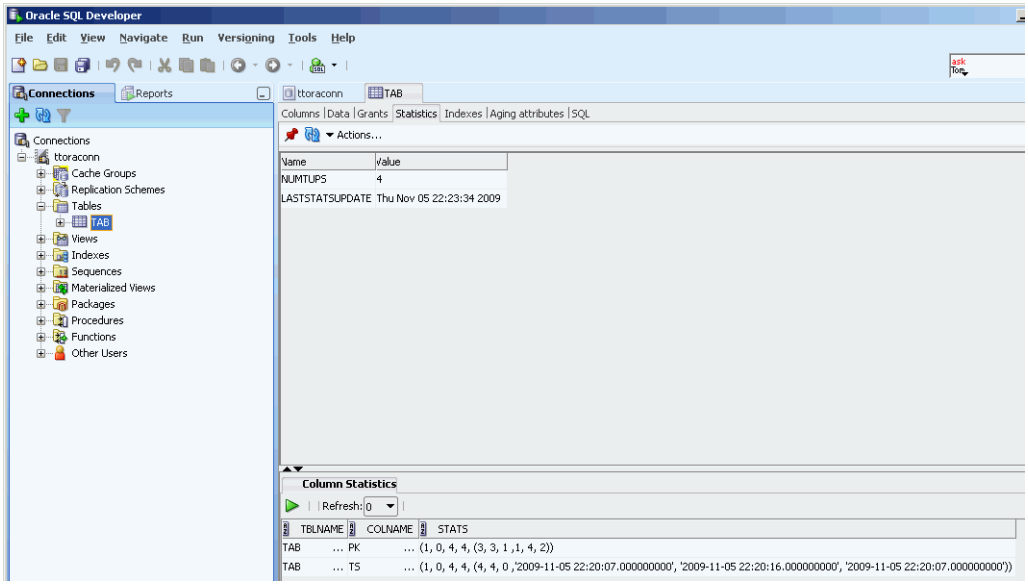
In the **Prompts** tab of the **Update statistics** dialog box, put a check in the **Invalidate referenced commands** check box if you want the invalidate the execution plans of statements that reference the table whose statistics will be updated. Invalidating a statement's execution plan causes that statement to be recompiled or reprepared upon its next execution.

In the **Interval type for table statistics** pull-down menu, select **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. A range index must be defined on the table in order to collect complete interval statistics.

Click the **Apply** button to update statistics for the table.



To view the optimizer statistics of a table, click the name of the table. If you are unable to find the table that you are looking for, see "Locate a TimesTen database element" on page 5-4 for information.



In the **Statistics** tab within the table's tab, 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, it shows:

- Its name
- The number of intervals the data was divided into in order 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, it shows:

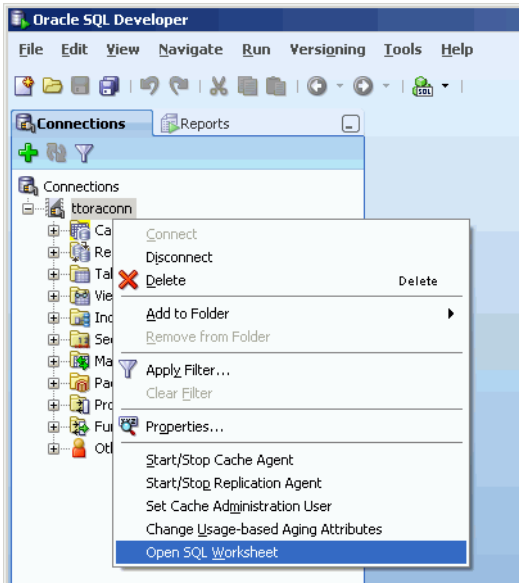
- The number of unique values other than the most frequently occurring value
- The number of rows that contain a value other than the most frequently occurring value
- The number of rows that contain the most frequently occurring value
- The minimum value
- The maximum value
- The most frequently occurring value

For more information about query optimizer statistics, see "Statistics" in the Oracle TimesTen In-Memory Database Cache Introduction, "When optimization occurs" in the *Oracle TimesTen In-Memory Database Operations Guide*, or "ttOptUpdateStats" in the *Oracle TimesTen In-Memory Database Reference*.

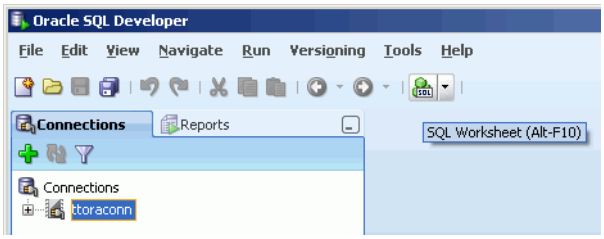
View the execution plan of a statement

Before a statement can be executed, it must first be prepared or compiled. The query optimizer generates an optimal execution plan for the statement when the statement is being prepared.

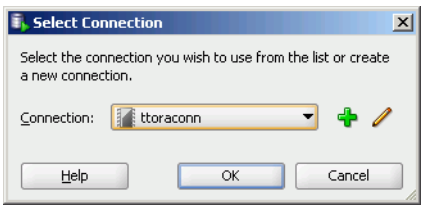
To view the execution plan of a statement, you must first type the statement in SQL Worksheet. If you do not have a worksheet open in your connection, right-click the node that contains the name of your database connection and select **Open SQL Worksheet**.



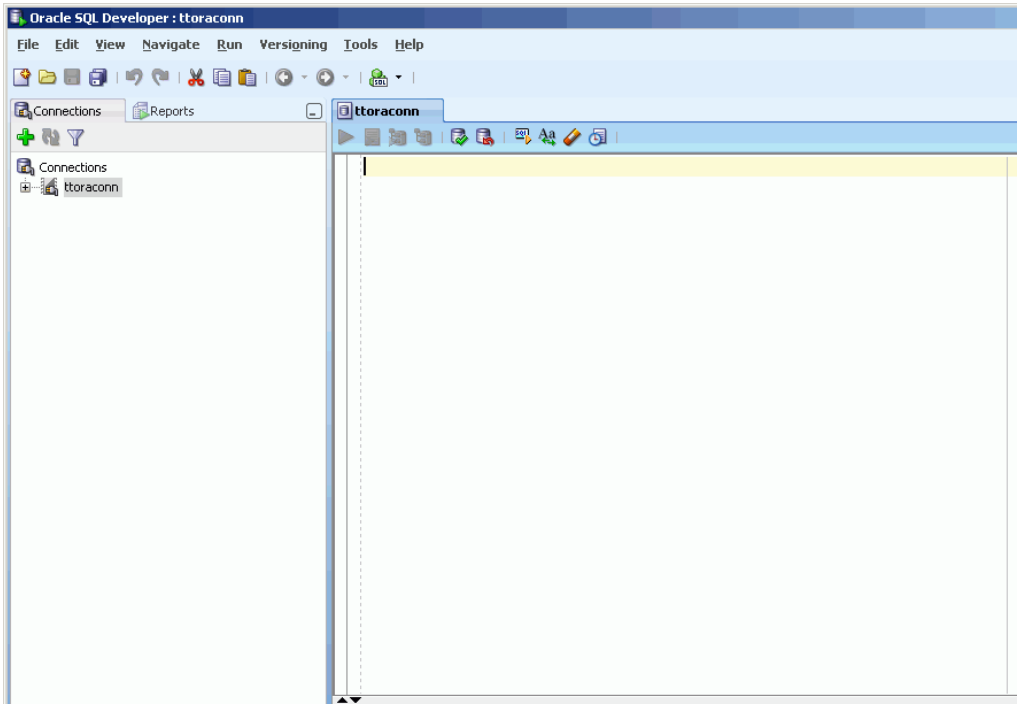
If you do not have a connection open, click the **SQL Worksheet** icon to establish a connection and then open a worksheet for that connection.



In the **Select Connection** dialog box, select the name of the connection in the **Connection** pull-down menu that you want to open a SQL Worksheet for, and then click the **OK** button.

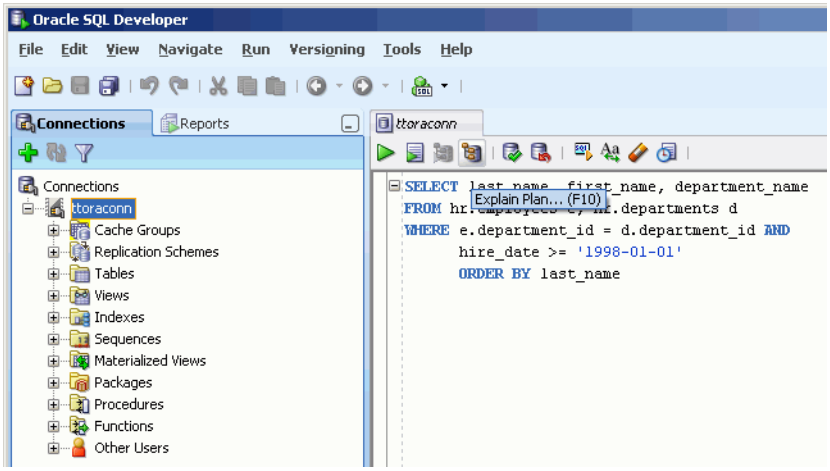


A connection to the TimesTen database will be established if a connection based on the selected connection name does not already exist.

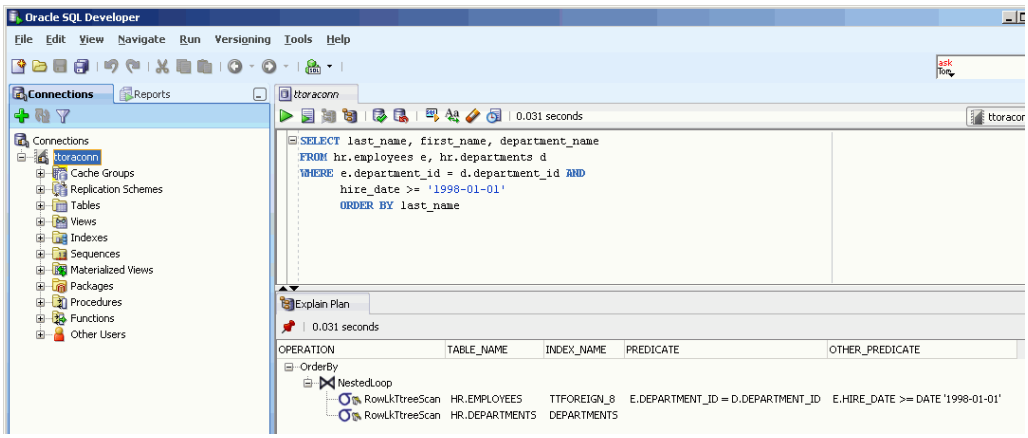


Before typing the statement in SQL Worksheet, make sure appropriate indexes have been created on the tables that are being referenced in the statement, and that table and column statistics have been updated among other performance tuning tasks. For more information about updating statistics, see "Update the table and column statistics used by the query optimizer" on page 7-1.

Type the statement in SQL Worksheet. Then click the **Explain Plan** icon to display the execution plan for the statement.



The execution plan is shown in the **Explain Plan** tab 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 which table or index the operation is performed on, or which expressions of the statement's predicate the operation is processing, if applicable. Types of operations that can appear in the execution plan include table or index scan (with lock granularity shown), join, group and sort.



For more information about interpreting an execution plan, see "Optimizer plan" in the *Oracle TimesTen In-Memory Database Cache Introduction* or "Viewing SQL query plans" in the *Oracle TimesTen In-Memory Database Operations Guide*.

For more information about query performance tuning, see "SQL tuning" in the *Oracle TimesTen In-Memory Database Operations Guide*.