

IDERA Comparison Toolset for Oracle Database 9.0

I D E R A

Exported on 08/09/2019

Table of Contents

Table of Contents.....	2
Quickly Compare and Safely Synchronize Oracle Databases.....	6
Release notes	7
New features and fixed issues	7
9.0 New features.....	7
Data Compare Command Line Wizard.....	7
Schema Compare Command Line	7
Schema Compare Command Line Wizard	7
Unified interface.....	7
Adds support for Oracle 18c and 19c	8
9.0 Fixed issues.....	8
General	8
Data Compare	8
Schema Compare.....	8
Known issues.....	8
Known issues.....	9
Welcome to Comparison Toolset for Oracle Database	10
What is IDERA Comparison Toolset for Oracle Database?	10
How Comparison Toolset for Oracle Database Helps.....	10
Getting started	11
Requirements.....	11
Cloud.....	11
Supported Oracle databases.....	11
9.0 Supported Oracle databases.....	12
Installing and uninstalling.....	12
Installing IDERA Comparison Toolset for Oracle Database	12
Uninstalling IDERA Comparison Toolset for Oracle Database	13
Updates and support.....	13

Activating a license	13
IDERA Schema Compare for Oracle Database.....	14
Schema Compare Application window	14
Schema Compare Application Settings	15
General	15
UI Preferences	16
Workspace	17
Schema Compare Workspace	18
Schema Compare Adding Oracle Connections	18
Schema Compare Adding Snapshots	20
Taking a snapshot.....	20
Adding snapshots to the workspace.....	20
Schema Compare Removing Oracle Connections	21
Schema Compare Entity Filters.....	22
Schema Compare Entity Filters Overview	23
Schema Compare Entity Filters Exclusion Rules.....	24
Schema Compare Working with Entity Filters.....	24
Schema Compare Entity Filter Conditions	25
Schema Compare Comparing Oracle Schemas	26
Schema Compare Schema Objects.....	26
Schema Compare Options.....	27
Schema Compare Comparison Options	28
Schema Compare Scripting Options.....	28
Schema Compare Execution Options	29
Schema Compare Comparison Results	30
Schema Compare Equal but Different	31
Schema Compare Refreshing Results.....	31
Schema Compare Synchronizing Schemas.....	31
Schema Compare Selecting Objects.....	32
Cleared but included	32
Schema Compare Generating the Synchronization Script.....	33

Schema Compare Working with the Synchronization Script	34
Warnings section	34
Schema Compare Executing Synchronization Script	35
Schema Compare Comparison Sessions	36
IDERA Data Compare for Oracle Database	38
Data Compare Application window	38
Data Compare Workspace	39
Data Compare Adding Oracle Connections	40
Data Compare Removing Oracle Connections	41
Data Compare Application Settings	42
Data Compare Comparing Data	44
Data Compare Comparing Oracle Schemas	45
Data Compare Mapping Tables Manually	45
Data Compare Comparison Options	46
Data Compare Scripting Options	47
Data Compare Preparing the Comparison	48
Data Compare Mapped Objects	48
Data Compare Mapping Rules	49
Data Compare Excluding Tables and Columns	50
Data Compare Comparison Keys	51
Data Compare Where Clauses	52
Data Compare Comparing the Data	53
Data Compare Synchronizing	55
Data Compare Generating Synchronization Script	56
Data Compare Working with the Script	57
Refreshing the comparison and regenerating the script	57
Data Compare Executing Synchronization Script	58
Data Compare Comparison Sessions	59
IDERA Data Compare command line	61
Data Compare Command Line Arguments Overview	61
Data Compare XML Configuration File Overview	62

Customizing Data Compare.....	64
Data Compare Modifying Comparison Options	65
Data Compare Changing Mapping Type.....	66
Data Compare Excluding Database Objects.....	66
Data Compare Excluding Columns	67
Data Compare Defining Custom Keys.....	67
Data Compare Defining Mapping Rules.....	68
Data Compare Specifying WHERE Clauses	68
Data Compare XML Configuration File Reference	69
Data Compare Error Codes.....	75
Comparison Toolset for Oracle Database PDF	76

Quickly Compare and Safely Synchronize Oracle Databases

- Compare and synchronize data and schema objects
- Easy navigation of user interface improves efficiency
- Automate comparison and synchronization operations
- Customize schema compare and synchronization sessions
- Generate ready-to-use database synchronization scripts

Release notes

IDERA Comparison Toolset for Oracle Database is a bundle of two products, IDERA Schema Compare and IDERA Data Compare. While Schema Compare allows you to compare Oracle database schemas and propagate schema changes from one environment to another, Data Compare lets you compare and synchronize tables across servers, databases, and versions when performing data migrations, copying databases, or auditing data.

To get a quick glimpse into the newest features, fixed issues, and known issues in this release of IDERA Comparison Toolset, review the following sections of the Release Notes:

- [New features and fixed issues](#)
- [Review issues fixed by this release](#)
- [See known issues](#)

New features and fixed issues

9.0 New features

Data Compare Command Line Wizard

The new Data Compare Command Line wizard simplifies the creation of the command line xml config file.

Schema Compare Command Line

The previous release of the Comparison Toolset for Oracle Database did not include a command line for the Schema Compare.

Schema Compare Command Line Wizard

Intended to simplify the creation of the command line config file.

Unified interface

A new and unified interface for all Oracle connection, used by schema, data and the command lines. The connection includes the following changes:

- Adds the option to use Oracle-specific connection settings.
- Includes the ability to check the validity of these settings.
- Removes support of the TNS connection type, mainly for simplicity.

Adds support for Oracle 18c and 19c

This release supports the new Oracle 18c and 19c versions.

9.0 Fixed issues

General

- The help file is updated, including a new section for the command lines.

Data Compare

- Data command line includes an option to encrypt Oracle credentials and the data source stored in the command line config file. The same option is supported by the schema command line as well.
- Data command line supports the UTF32 file encoding. The previous UTF7 is removed.
- Data Compare script no longer drops disabled triggers or foreign keys. Disabled constraints have no impact on the synchronization process.
- Data command line no longer requires the schema in the object name.
- Data Compare now sets the transaction isolation level to **READ COMMITTED**. This level works much better than the previous setting, which was **Serializable**. **READ COMMITTED** is also the Oracle default value.

Schema Compare

- Schema filters are simplified. These filters are intended to provide object exclusion. Initially, filters were advertised to achieve both inclusion and exclusion, which caused confusion.
- Schema Compare no longer collects information about tablespaces, data files, or temp data files. The Oracle views that contain this information require admin permissions and fail in most cases.
- Schema Compare now collects the current user and the user schema from the user context. The Oracle views used in the previous version needed admin permissions.
- A fix in the trigger script now shows the trigger status in Schema Compare. This information is useful when the triggers are disabled.
- A change in the definition parsing of views, procedures, functions, packages, types, and triggers fixes some rare issues related to scripting of these objects.
- Schema Compare now supports the schema triggers, similar to the DDL trigger on SQL Server.

Known issues

IDERA strives to ensure our products provide quality solutions for your database needs. The following known issues are described in this section. ***If you need further assistance with any issue***, please contact [IDERA Support](#).

Known issues

There are no known issues in this release.

Welcome to Comparison Toolset for Oracle Database

IDERA Comparison Toolset for Oracle Databases provides database administrators and developers the tools they need to safely and efficiently manage the database change process.

This wiki contains useful information that will help you get started and take full advantage of the best database comparison and synchronization tools for Oracle in the market. The products covered in this wiki include:

- [IDERA Schema Compare for Oracle Database](#)
- [IDERA Data Compare for Oracle Database](#)
- [IDERA Data Compare Command Line](#)

What is IDERA Comparison Toolset for Oracle Database?

Data schema and data comparison is sometimes an extremely time consuming and confusing task when completed manually, and the process can be prone to error. Rather than spending hours digging through and scrutinizing data and the overall database structure, then manually generating scripts, wouldn't it be much easier if there was a tool to automate this process?

IDERA Comparison Toolset for Oracle Database provides an easy, automated way to compare schema and data, generate synchronization scripts, and deploy those scripts to synchronize databases with your SQL Server and Oracle Databases. Whether migrating changes from test to production or repairing unwanted changes, IDERA Comparison Toolset for Oracle Database saves you time and reduce the headaches of performing manual database comparisons mistakenly applied and unwanted changes.

How Comparison Toolset for Oracle Database Helps

When you need to quickly identify and synchronize data and object changes, you can use IDERA Comparison Toolset for Oracle Database to:

- Perform Data Compare and Schema Compare tasks and synchronization across your SQL Server and Oracle Databases
- Capture schema snapshots to preserve change history
- Synchronize SQL data automatically via command-line
- Customize your environment using workspaces and saved comparison options

Getting started

When you download the IDERA Comparison Toolset for Oracle Database from our web site, you receive an .msi file that you can run to install the comparison package.

The installation set up adds the IDERA Comparison Toolset for Oracle Database group in the Start menu as well as places shortcuts of the IDERA Schema Compare for Oracle Database and IDERA Data Compare for Oracle Database on the desktop.

For three weeks from the installation date, the products are fully functional and have no restrictions. After the trial period expires, you must acquire a license. You can purchase a license from our website at IDERA.com.


The Getting Started section contains the following important topics necessary for rolling out and maintaining the product.

- [Requirements](#)
- [Supported Oracle databases](#)
- [9.0 Supported Oracle databases](#)
- [Installing and uninstalling](#)
- [Updates and support](#)
- [Activating a license](#)

Requirements

Minimum requirements for running IDERA Comparison Toolset for Oracle Database are:

- .NET Framework 4, available from the [Microsoft Developer Tools Download Center](#)
- at least 50MB of free space on the hard drive where the application will be installed

 Depending on the size of the schemas being compared, data and schema operations may require significant amount of disk space.

- Windows XP SP3 or later / higher versions of Windows operating systems
- minimum of 512MB of memory; 1GB or higher recommended
- minimum resolution of 1024x768 suggested for the GUI

Cloud

IDERA Comparison Toolset for Oracle Database runs on cloud virtual machines (such as Microsoft Azure VM and AWS EC2) with Microsoft Windows. It supports Oracle Database instances on cloud virtual machines (such as Microsoft Azure VM and AWS EC2).

Supported Oracle databases

IDERA Schema Compare for Oracle Database supports the following Oracle databases:

- Oracle 12c
- Oracle 11g
- Oracle 10g

IDERA Data Compare for Oracle Database supports the following Oracle databases:

- Oracle 12c
- Oracle 11g
- Oracle 10g
- Oracle 9i

9.0 Supported Oracle databases

IDERA Schema Compare for Oracle Database supports the following Oracle databases:

- Oracle 19c
- Oracle 18c
- Oracle 12c
- Oracle 11g
- Oracle 10g


IDERA Data Compare for Oracle Database supports the following Oracle databases:

- Oracle 19c
- Oracle 18c
- Oracle 12c
- Oracle 11g
- Oracle 10g
- Oracle 9i

Installing and uninstalling

Installing IDERA Comparison Toolset for Oracle Database

You can download the IDERA Comparison Toolset for Oracle Database from our website at www.idera.com. The installation setup will add the **IDERA Comparison Toolset for Oracle Database** group in the Start menu as well as place shortcuts of IDERA Schema Compare for Oracle Database and IDERA Data Compare for Oracle Database on the desktop.

 Generally, new builds of the comparison tools can be installed on top of the previously installed builds without having to uninstall first.

Uninstalling IDERA Comparison Toolset for Oracle Database

There are two processes you can use to uninstall IDERA Comparison Toolset for Oracle Database. Use *Process 2* if you still have the original installation package.

To uninstall IDERA Comparison Toolset for Oracle Database

Process 1

1. Go to Control Panel, and then click **Programs > Uninstall a program**.
2. Click **IDERA Comparison Toolset for Oracle Database**, and then click **Uninstall/Change**.
3. Follow the prompts to complete uninstallation.

Process 2

1. **If you still have the original installation package**, double click the icon, and then click **Remove**.
2. Follow the prompts until the software is removed from your machine.

Updates and support

We update our products periodically but will only notify our customers via email when we publish an update that contains a critical fix or that is considered a major upgrade. Therefore, to ensure that you are always running the latest version of our products, we recommend that you check our site, www.idera.com, regularly for product updates.

Activating a license

If you have acquired a license, you can quickly activate. Go to **Start > All Programs > IDERA > Comparison Toolset for Oracle Database > License Activation**. In the License Activation window, type or paste the license number you received, and then click **Activate License**. The license activate utility will connect to a web service running on our web servers that will validate the license you provided. Within a few seconds you should see a confirmation message indicating that the product was successfully upgraded.

IDERA Schema Compare for Oracle Database

Welcome to IDERA Schema Compare for Oracle Database, the tool of choice for thousands of DBAs and developers around the globe. Identifying schema differences between two Oracle databases and synchronizing those schemas is easy, fast, and safe with IDERA Schema Compare for Oracle Database.

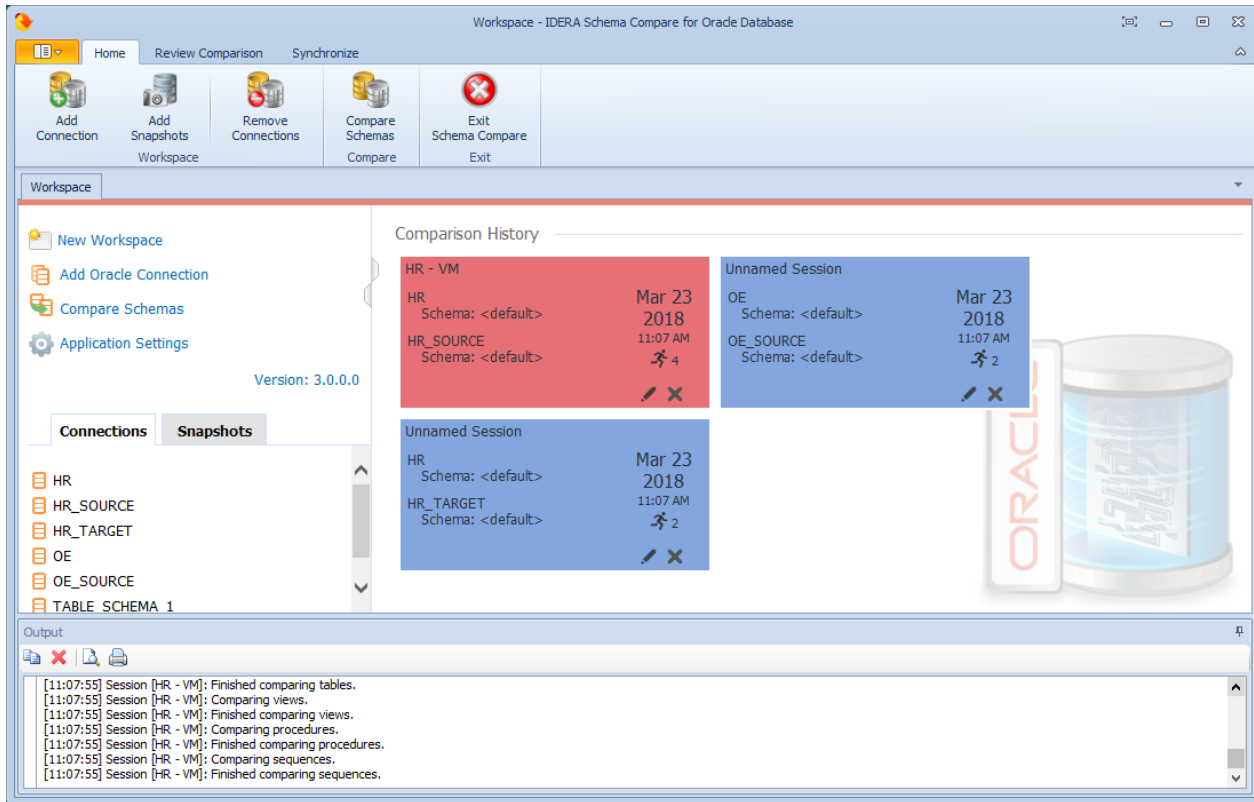
This release IDERA Schema Compare for Oracle Database is built on top of a new, robust, efficient and scalable comparison framework.

Please direct any questions, comments and suggestions about the IDERA Schema Compare for Oracle Database to support@idera.com.

Schema Compare Application window

IDERA Schema Compare for Oracle Database divides the main application window into three areas:

1. **The ribbon.** The context sensitive ribbon provides all the command buttons and icons you need to do your job quickly. The ribbon is organized in three distinct tabs each of which corresponding to the main steps of the database schema comparison and synchronization process and containing the relevant command buttons.
2. **Main panel.** The main panel contains a permanent tab called the [Workspace](#) which you cannot close, and a tab for each active comparison session.
3. **Output window.** The output window provides a running log of the tasks being performed in IDERA Schema Compare for Oracle Database.



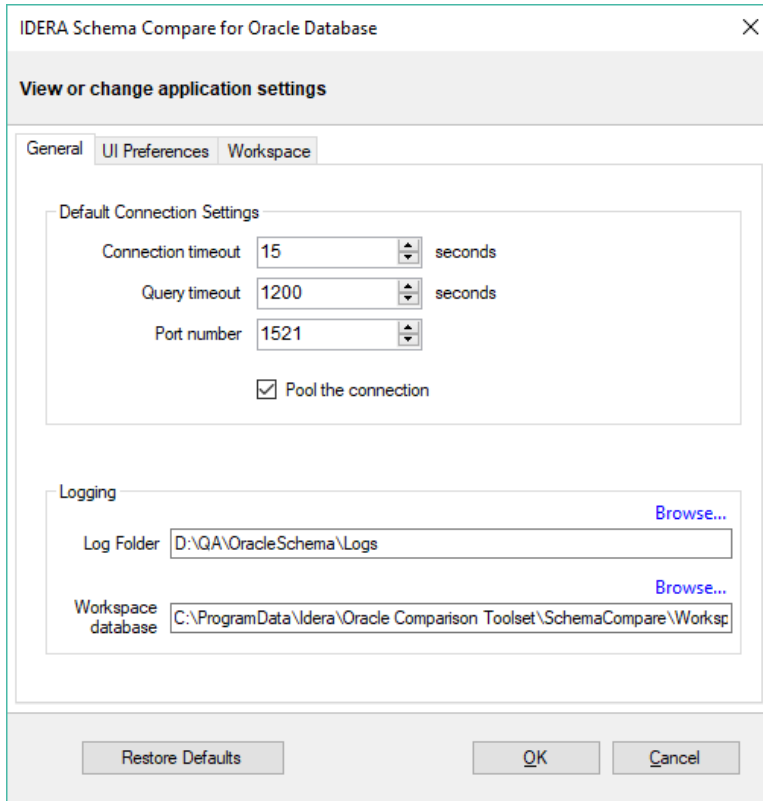
Schema Compare Application Settings

Application settings allow you to tweak the behavior of IDERA Schema Compare for Oracle Database. It can be accessed from the drop-down menu off of the top-left corner product icon or from the **Application Settings** link on the Workspace.

Application level settings are grouped into the following tabs:

General

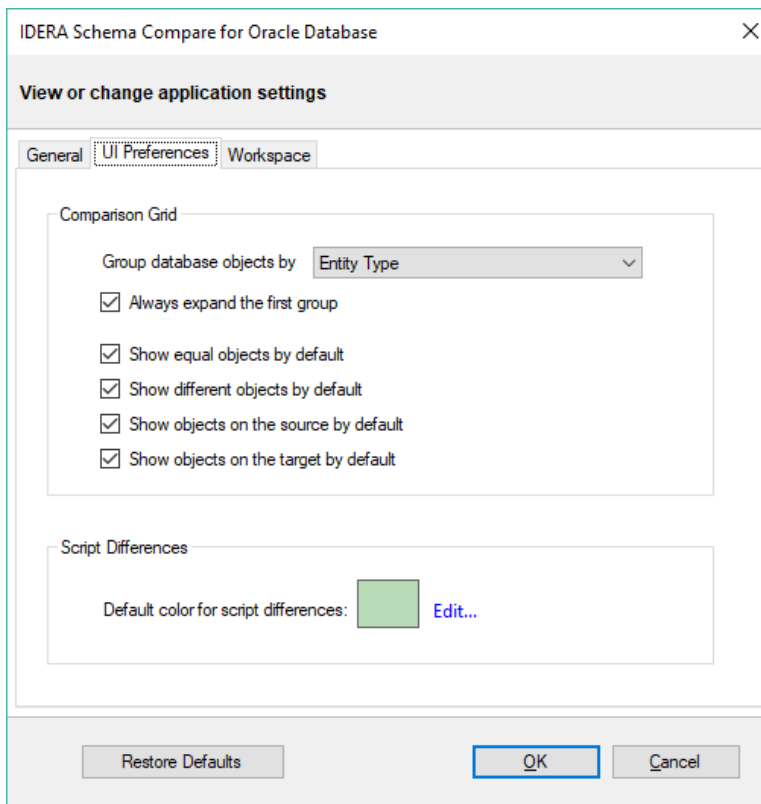
- **Default Connection Settings.** Specifies the **Connection timeout**, **Query timeout**, **Port number**, and whether the connections are pooled. Under normal circumstances, you do not need to change any of these settings.
- **Log folder.** Specifies the location where logs are stored. The user running the application must have write access to that location.
- **Workspace database.** Specifies the path and the name of the workspace database file. You can change this field by clicking on **Browse**, and then selecting a custom file.



UI Preferences

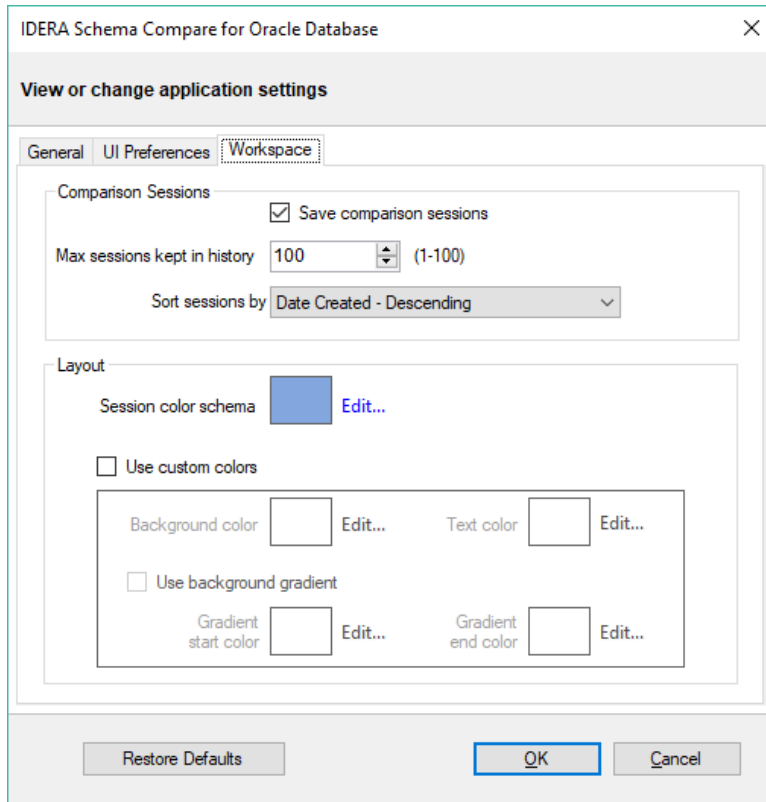
- **Comparison Grid.** Allow you to adjust the behavior of the comparison results grid based on your preferences.

- **Script Differences.** Allow you to change the highlight color for the script differences.



Workspace

- **Comparison Sessions**
 - **Save comparison sessions.** Selected by default. It instructs the IDERA Schema Compare to save comparison sessions for future use. A saved session contains all the necessary information so that you can rerun the comparison with one click. Saved sessions appear on the main panel of the Workspace.
 - **Max sessions kept in history.** By default, IDERA Schema Compare keeps the last 100 comparison sessions. Although there is no noticeable performance difference on application launch related to the number of sessions stored, you may choose to set this parameter to a lower number. Allowable values are 1 to 100.
 - **Sort sessions by:** Change the order in which stored sessions appear on the Workspace.
- **Layout.** Change the color schema of a session. You can select from built-in color schemas or pick custom colors.



Schema Compare Workspace

IDERA Schema Compare for Oracle Database Workspace is the storage and the interface for managing Oracle connections, credentials, comparison sessions, and other UI options.

Schema Compare Adding Oracle Connections

Before comparing Oracle databases, you must add Oracle connections to the Workspace. The **Add Connection** functionality is accessed from the ribbon (the **Home** tab) or from the **Add Oracle Connection** link at the top left section of the Workspace.

On the **Add Connection** window, you can specify:

- **Connection Name.** A name for the Oracle connection.
- **Authentication type.** Indicates whether you want to use Oracle or operating system authentication. If **Oracle authentication** is specified, you also must complete the **Username** and **Password** fields.
- **Role.** Indicates the set of privileges associated with the connection. You can choose default, SYSDBA, or SYSOPER.

In the **Basic Settings** tab, you can choose the connection type and specify settings required by the connection. There are four connection type you can choose from:

- **Basic Connection.** Requires the name or IP of the host system for Oracle database, the port number and the service name. If the port number is **0**, the default port **1521** is used.
- **TNS Identifier.** Requires a connect identifier that exists in the Oracle **tnsnames.ora**. You cannot use this type of connection to establish a connection to a remote Oracle database. The file **tnsnames.ora** must exist on the local system.
- **Easy Connect.** Requires an easy-connect string for the Oracle database. The format of the easy-connect string is `username/password@[//]host[:port][/]service_name`. Because the username and password are already included in the easy-connect string, the credentials specified at the top section of the **Add Connection** form are ignored when the easy-connect option is selected.
- **Service Descriptor.** Requires a connect descriptor in the format:

```
(DESCRIPTION =
  (ADDRESS_LIST =
    (ADDRESS =
      (PROTOCOL = TCP)(Host = <hostname>)(Port = <port>)))
    (CONNECT_DATA = (SERVICE_NAME = <sid>))
  )
```

The **Test Connection** link allows you to verify whether the settings are correct and the connection to Oracle database can be established. You can add Oracle connection to the workspace by clicking **OK**.

Schema Compare Adding Snapshots

A schema snapshot is a compact file that contains all the schema information for a database (connection). The schema information is stored in a proprietary format and can be read only by IDERA Schema Compare for Oracle Database. The schema snapshot does not contain data, only schema information.

Schema snapshots allow you to maintain a history of schema changes. You can take schema snapshots at different points in time and then you can compare those snapshots to each other or compare the snapshots to the live database to get a clear and accurate picture of the "evolution" of the database schema.

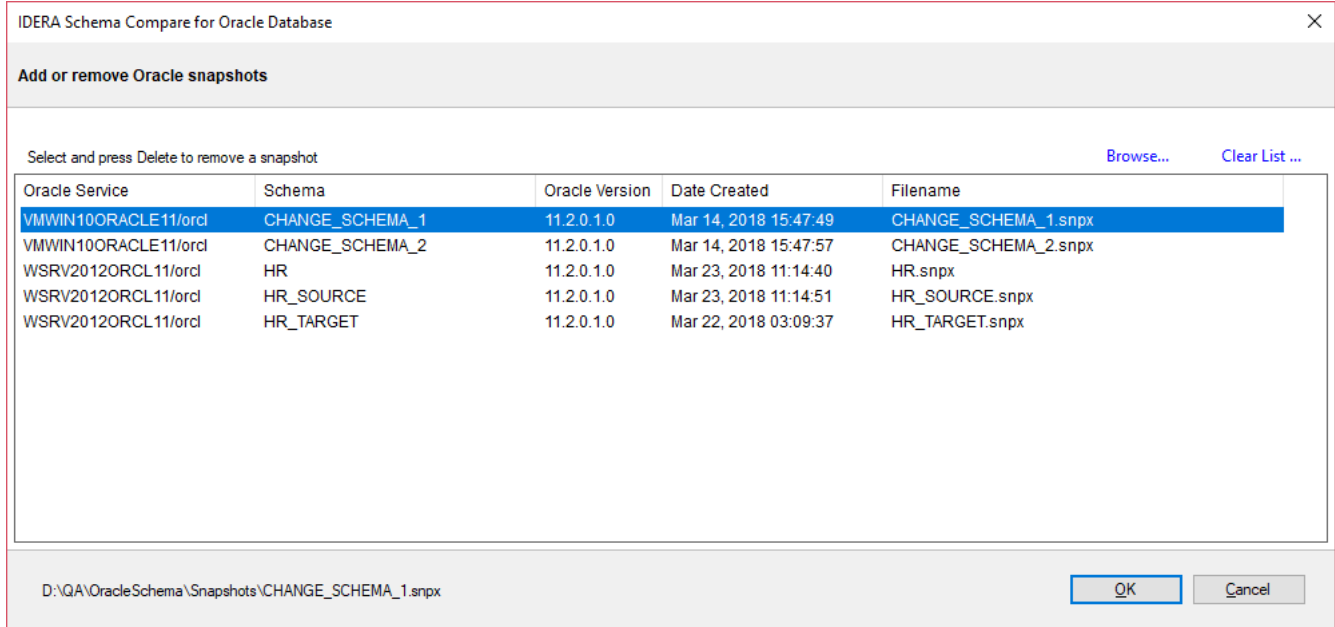
Taking a snapshot

To take a schema snapshot, simply move the mouse over the name of the database (connection) on the left panel of the Workspace and you will see a camera icon appear on the side of the database name. Click the icon to take the snapshot. Choose the name of the snapshot and the location where you wish to save it.

Adding snapshots to the workspace

Before comparing snapshots to each other or to live databases you must add the snapshots you need to the workspace. The **Add Snapshots** functionality can be accessed from the ribbon (the Home tab).

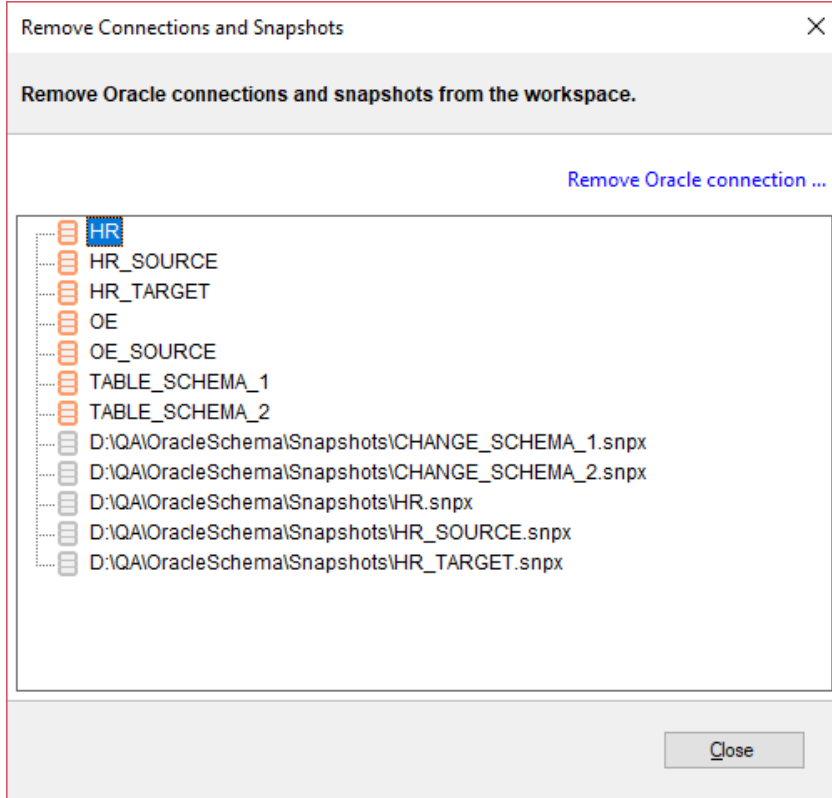
On the Add Snapshots window, click **Browse...** to browse to the location where the snapshot(s) you wish to add to the workspace are stored. You can select as many snapshots as you want from different locations. For each snapshot you select the application displays the file name, the database the snapshot was taken from, the date when the snapshot was taken, and Oracle version.



Schema Compare Removing Oracle Connections

There are two ways to remove an Oracle connection from the workspace:

- Move the mouse over the name of the connection you wish to remove under the **Registered Oracle Connections** on the left panel of the Workspace. The connection name is highlighted and a **Remove** link appears.
- Click **Remove Connections** on the Home tab of the ribbon. A window appears showing all the registered connections and snapshots. Select the connection you wish to remove, and then click **Remove Oracle connection** at the top right corner. Confirm the removal when prompted.



Schema Compare Entity Filters

Entity filters allow you to include or exclude from the comparison certain objects based on some predefined criteria. Those filters are helpful when you are dealing with very large schemas and wish to compare only a small subset of the objects.

You can access the entity filters using one of the following areas:

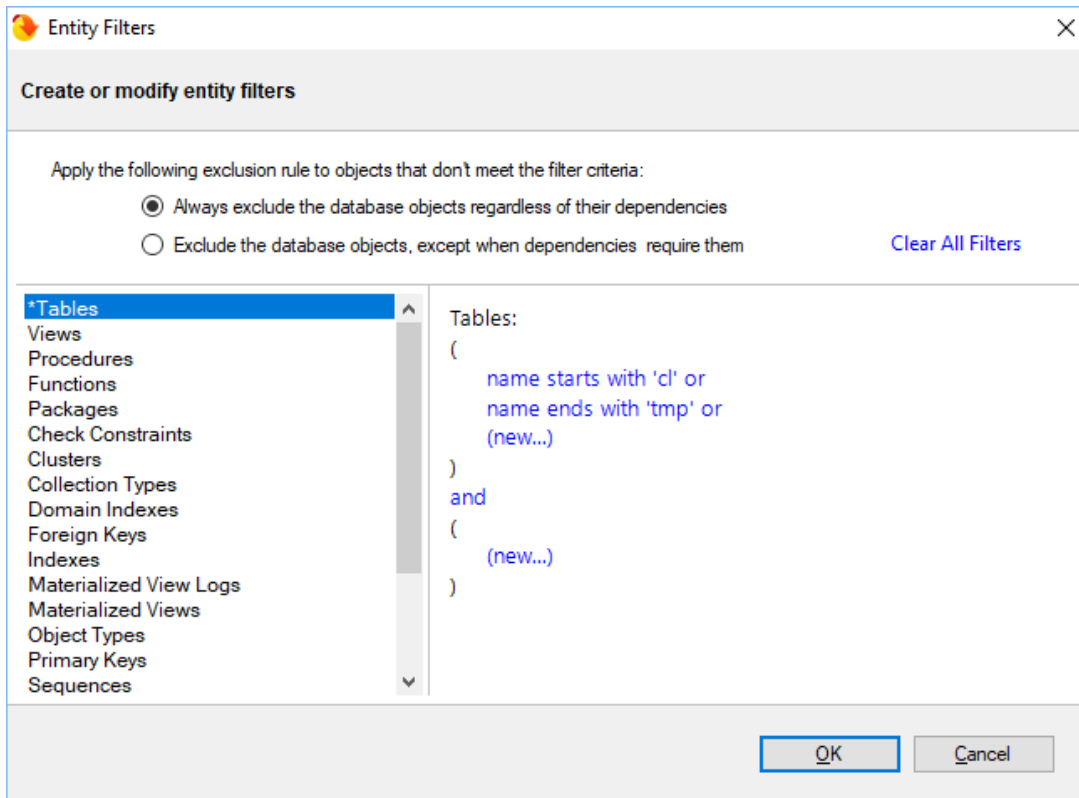
- **From the Select the Oracle Schemas to Compare window.** This area allows you to exclude the objects you want before the comparison occurs.
- **From the ribbon or context menu after the comparison is complete.** Setting entity filters at this stage causes the comparison to repeat. Once you set the filters, they are stored on the comparison session so the next time you run the same comparison, those filters are applied.

To learn more about entity filters, see the following topics:

- [Entity filters overview](#)
- [Creating exclusion rules](#)
- [Working with entity filters](#)

Schema Compare Entity Filters Overview

The entity filter is a collection of conditions organized into units called groups. A single group contains conditions combined with the same operator: AND/OR. The groups themselves are combined with an operator that can be the same or different from the conditions' operator.



There is no limit to the number of groups that a filter can have or the number of conditions contained in a group. To keep the filter simple, we recommend that you place all conditions that can be combined with a common operator in the same group. Create additional groups only if you need conditions that require different operators.

The following filter, for example, defines two conditions combined with the AND operator. It requires only one group.

- name does not start with "v_temp" and
- name does not contain "temp"

The following filter contains two conditions combined with the AND operator and requires only one group:

- name does not start with "v_temp" and
- name does not contain "temp"


The following table filter, on the other hand, contains conditions that require different operators and two groups are needed:

Group 1:

- name starts with "emp" or
- name ends with "emp"

Group 2:

- name does not start with "tmp" and
- name does not start with "#" and
- name does not start with "##"

 An entity filter, similar to other comparison settings, is saved in the comparison session and it is applied automatically when you launch the session.

Schema Compare Entity Filters Exclusion Rules

An entity filter provides two options, located at the top of the filter form, that determine the method by which database objects are excluded:

- The first option always excludes objects that do not meet the filter criteria. The excluded objects do NOT appear in the comparison grid at all.
- The second option excludes objects that do not meet the filter criteria, but allow the database dependencies to include them (if these objects are needed). Excluded objects in this case appear unchecked in the comparison grid.

Please note that the second exclusion rule could produce, in some cases, unexpected results. You might notice that database objects, explicitly excluded by the filter, still appear in the synchronization script.

Consider, for example, the following scenario:

A database contains a table named "employees" and a view named "v_employees" that depends on it. A table filter is defined that excludes the "employees" table. Since the view "v_employees" depends on the "employees" table, the "employees" table is included if the view needs to be synchronized.

If you wish to always exclude database objects that do not meet the filter criteria, regardless of the dependencies, do one of the following:

- Select the first exclusion rule OR
- Leave the second rule selected, but go to the Comparison Options form and uncheck the option, **Always synchronize dependencies of user-selected objects.**

The first option is recommended.

Schema Compare Working with Entity Filters

The conditions for an entity filter can be created or modified via the links provided in the filter definition section.

To create a new condition

1. From the list of objects, select the object type to which the filter applies.
2. Click on the (new...) link in the filter definition.
3. Enter the condition criteria, and then click **OK**.

i Schema object types with an associated filter have a star * in front of the name and appear at the top of the list.

To modify or delete an existing condition

1. In the filter definition, click on the condition that you wish to modify.
2. In the condition form, change the condition's property, and then click **OK** or click **Delete** to remove it.
3. To change an operator, simply click on it and the operator flips from AND to OR or vice versa.

Schema Compare Entity Filter Conditions

The conditions are the basic filter criteria that are evaluated to determine whether an object should be included or excluded. A filter condition exposes the following properties:

- **Name/Schema.** Indicates whether the condition is checked against the name or the schema of the database object.
- **Condition type.** Determines the type of the condition, such as "Starting With," "Equals To," etc.
- **Case sensitive.** Indicates whether the condition should honor the character case.
- **Condition expression.** Defines the condition expression.

Most of the conditions types are straightforward. The filter supports also two conditions that are based on regular expression for more complex criteria.

The following condition, for example, selects the database objects that have at least one digit in their name:

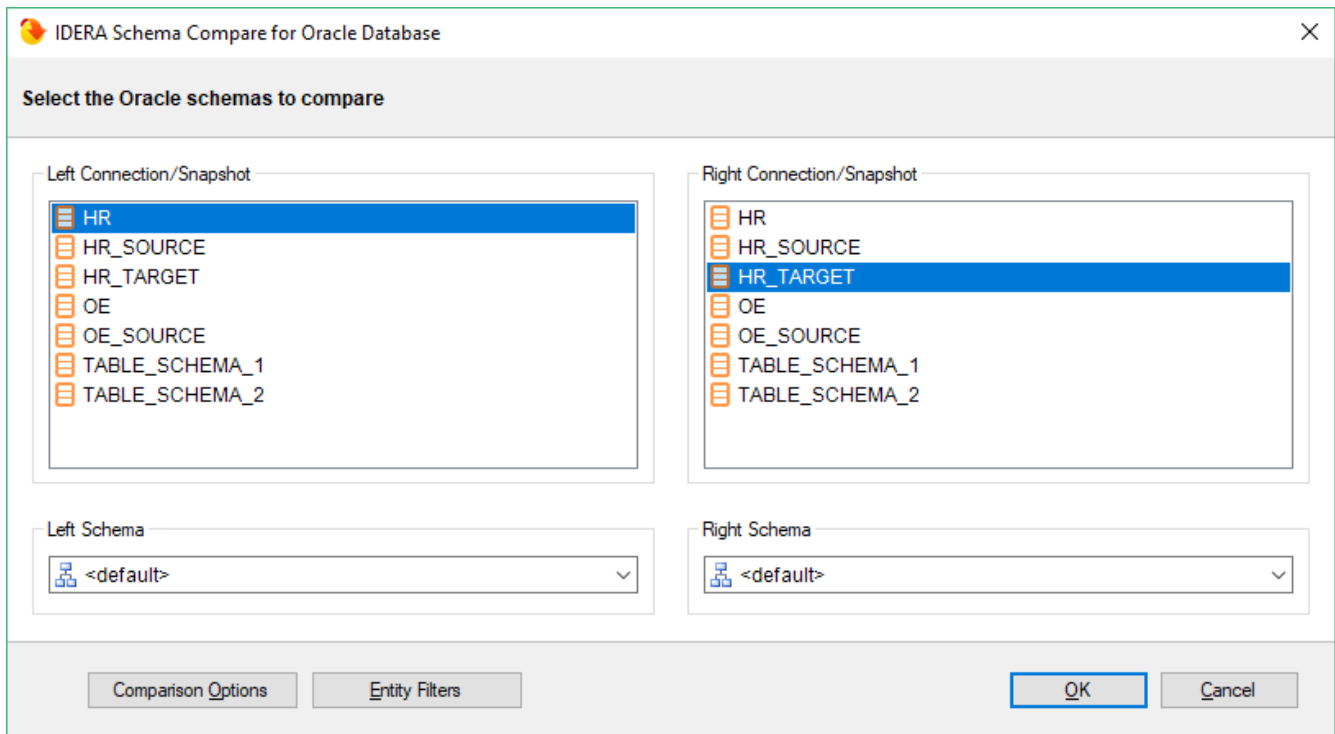
The screenshot shows a dialog box titled "Filter Condition" with a close button (X) in the top right corner. The dialog contains the following elements:

- "Apply condition to": A dropdown menu with "Name" selected.
- "Condition type": A dropdown menu with "Match a Regular Expression" selected.
- "Case sensitive": An unchecked checkbox.
- "Condition expression": A text input field containing the regular expression "\d+".
- Buttons at the bottom: "Delete", "OK", and "Cancel".

Schema Compare Comparing Oracle Schemas

To start a comparison between two databases, click **Compare Schemas** on the Home tab of the ribbon. The Select Oracle Schemas to Compare dialog appears, allowing you to choose the connections where the schemas you wish to compare reside as well as the schemas themselves.

From this window, you can also access the Comparison Options that are applied during the comparison as well as the entity Filters described in the **Entity Filters** section.



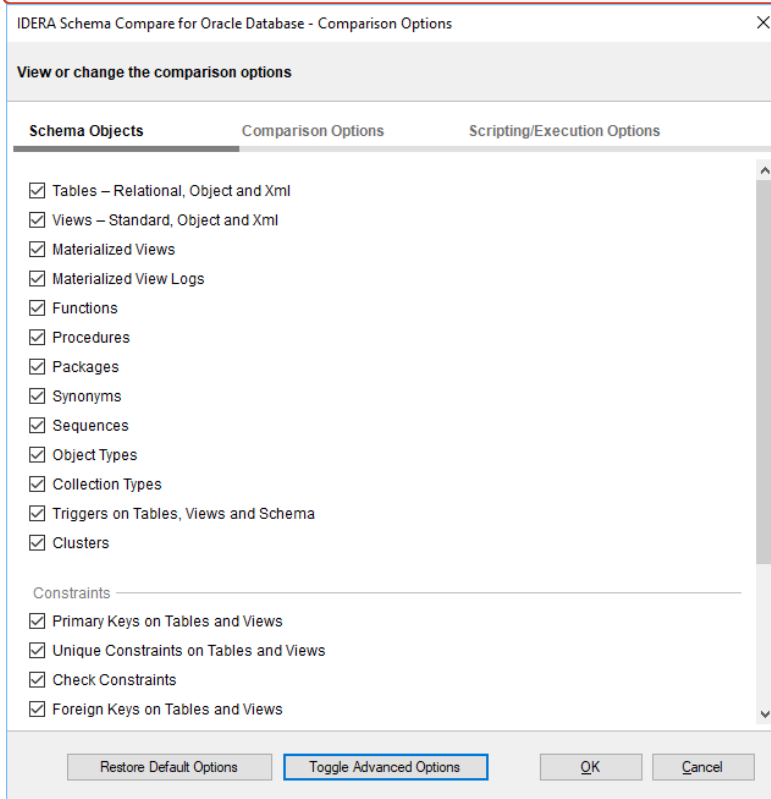
Schema Compare Schema Objects

IDERA Schema Compare for Oracle Database supports over 20 different schema object types. By default, all of the object types are included in the comparison and synchronization.

In the majority of cases, you should not need to change the default settings. There are scenarios in which you might wish to exclude certain categories that are included by default.

To exclude or include certain object types, click **Comparison Options**. The first tab on the Comparison Options window is the Schema Objects tab which allows you to select or clear the object types you want.

⚠ Because of the dependencies between database objects, excluding all objects of a given type may result in a synchronization script that fails to execute due to invalid dependencies. For more information about the handling of dependencies, see [Schema Compare Comparison Options](#).



Schema Compare Options

IDERA Schema Compare for Oracle Database exposes over 50 different options that provide the user with highly granular control of the behavior of the comparison and synchronization engine.

The defaults were carefully selected to match the most common scenarios so that in the majority of the cases you don't even need to look at those options at all, however, every Oracle implementation is different and there is no "one size fits all" scenario.

For clarity and easy access, the options have been organized on three different tabs:

- [Schema Compare Comparison Options](#)
- [Schema Compare Scripting Options](#)
- [Schema Compare Execution Options](#)

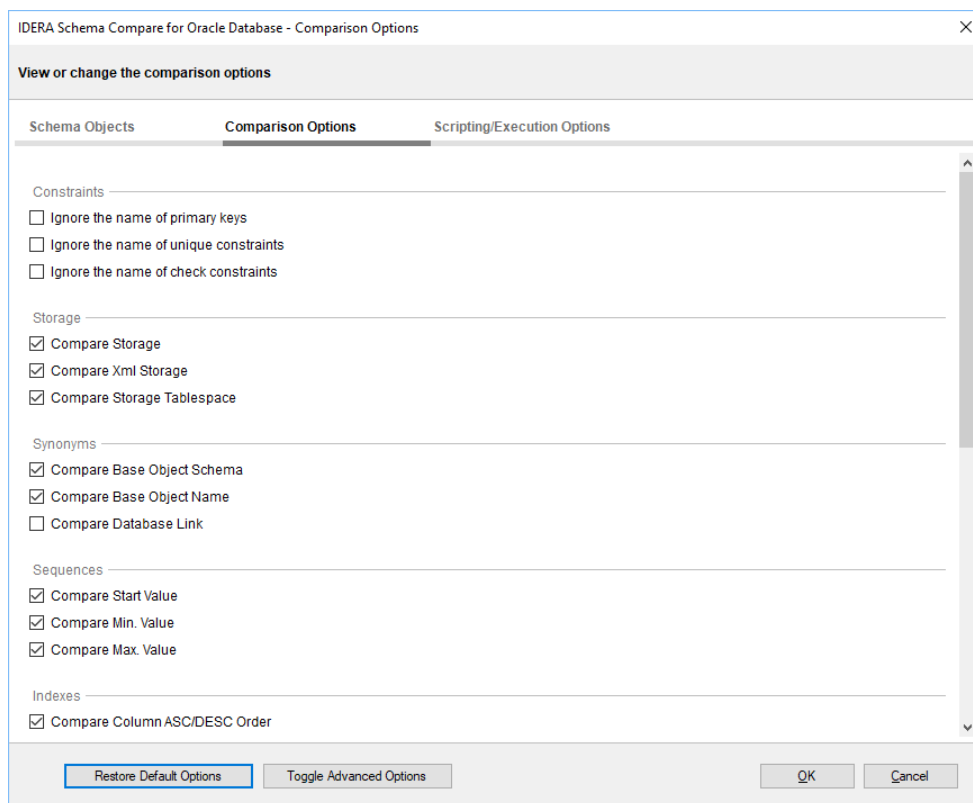
Note that the two advanced options tabs are not visible by default, but you can show or hide them by clicking **Toggle Advanced Options**.

Schema Compare Comparison Options

When are two database objects equal? The answer is: it depends! It depends on what you want to compare those objects. For example: is "Equal" equal to "equal?" The comparison options allow you to have complete control over the comparison engine which considers those options before deciding to mark a pair of objects as equal or different.

So, when two object are found to be different by the comparison engine while you believe they should be equal, or they are found to be equal while you believe they should be different don't panic. Investigate the differences and carefully review the comparison options. More likely than not one of those options holds the answer to your surprise.

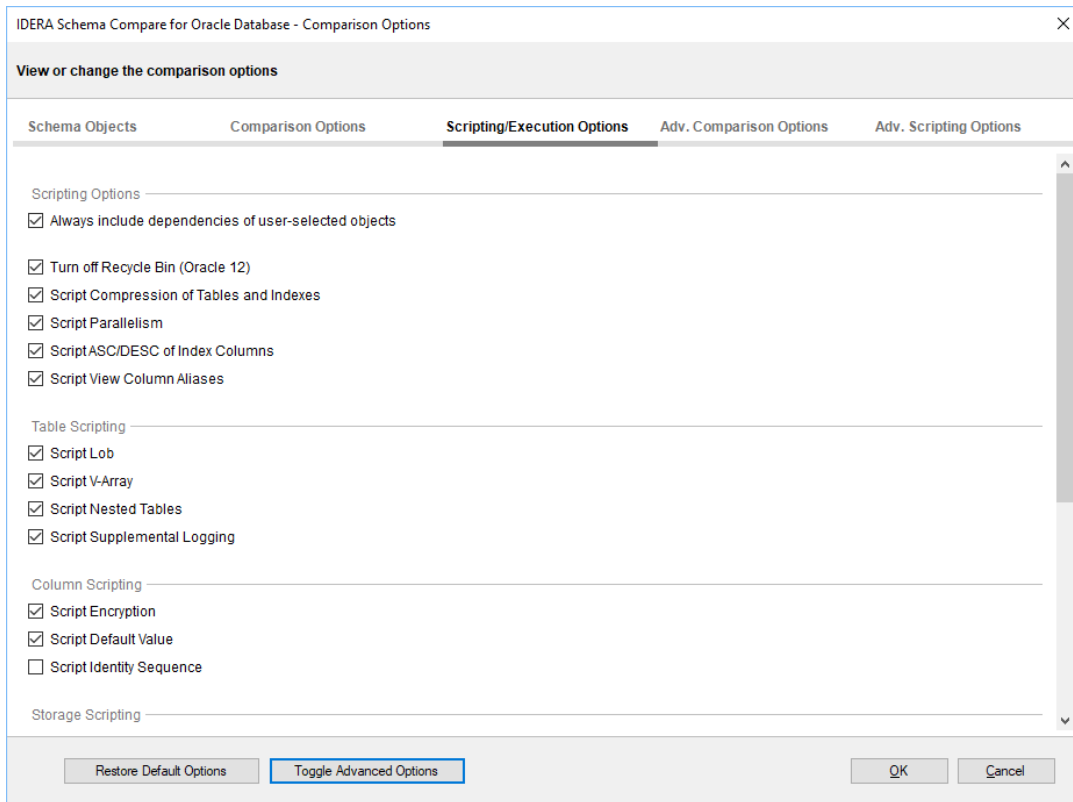
Please do note that some of the options, especially those listed on the advanced tab may be unknown to you depending on the level of familiarity with Oracle and it would have been helpful to provide a description for each. However, due to space and other restrictions, instead of us providing an incomplete and insufficient description that attempts to summarize information already available in the Oracle documentation we selected to omit those descriptions and let you consult the source directly when necessary.



Schema Compare Scripting Options

By default the IDERA Schema Compare for Oracle Database includes almost all elements in the synchronization script, however, a large number of scripting options allow the user to exclude any element they want. For example the user may decide that h/she does not want to script parallelism, or column encryption, or Xml Storage etc.

The scripting options are self-explanatory. A checked option indicates that that particular element is scripted whereas, un-checking or clearing the option causes the exclusion of that element from the synchronization script. Similar to Comparison Options, the Scripting Options are separated into two tabs, the first one contains the more common options and the second tab (Advanced Scripting Options) contains less common options that most users would not need to change.



Schema Compare Execution Options

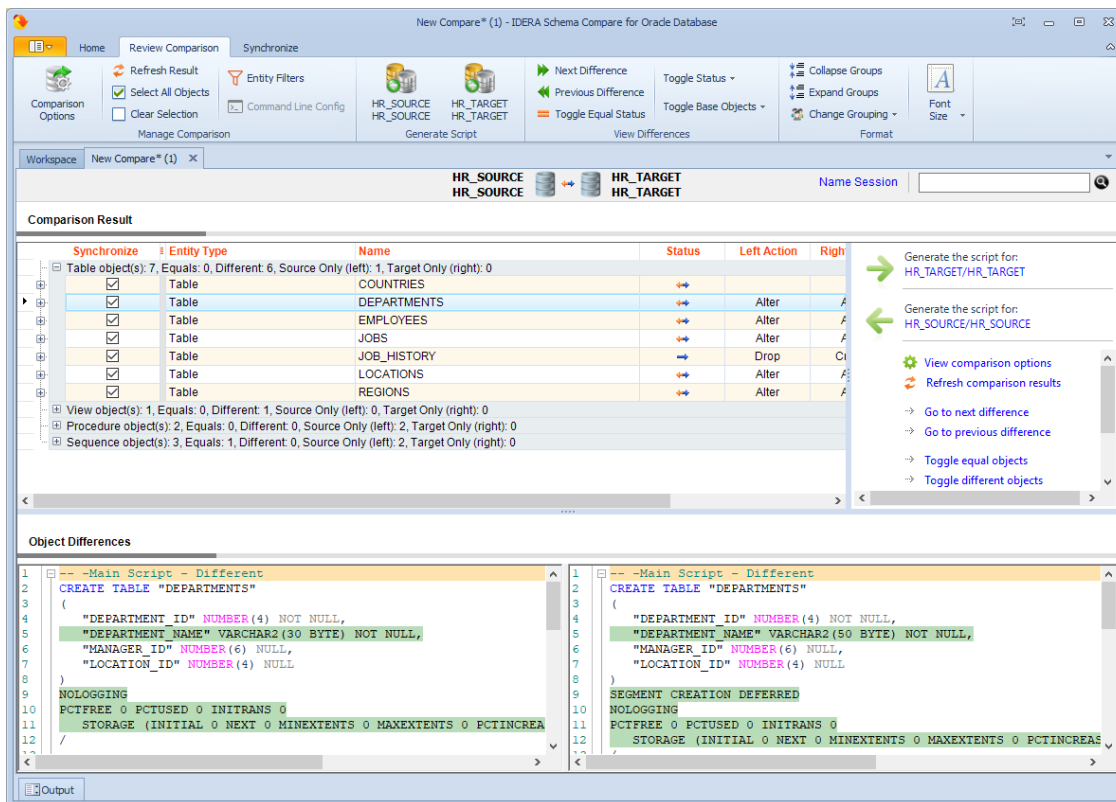
There are only two execution options available and those are listed at the bottom of the Scripting/Execution Options tab:

- **Continue execution upon error.** By default, if possible, the IDERA Schema Compare for Oracle Database attempts to continue the execution of the synchronization script upon encountering an error.
- **Log script execution.** By default, an execution log is created and stored at the location defined in the application settings. If the execution fails, you may consult the log for the error details.

Schema Compare Comparison Results

Upon completion of the schema compare operation, the results of the comparison are displayed on a new comparison session tab. The Comparison Results window is divided into three main sections:

1. **Comparison summary.** The main section displays the list of all objects that were compared, grouped by object type, indicating for each object the result of the comparison (equal, different, exists only on one of the databases). By default, all the objects are displayed including the ones that are identical, however, a handy context menu allows you to filter objects based on object type or/and based on the result of the comparison.
2. **Differences.** The bottom section of the results window displays, side by side, the scripts of the object pair that is selected in the top section of the results grid in both of the schemas being compared. The differences are highlighted using a simple color coding scheme.
3. **Action.** The right panel provides quick access to action buttons/links that allow you to generate synchronization scripts, view/change comparison options, refresh comparison results etc.



Schema Compare Equal but Different

Making sense of the comparison results

Columns are marked as different but they seem to be identical! In some cases you may see a table pair marked as different and when you look at the "Schema Differences" tab on the bottom you may notice that one or more columns have been highlighted as being different but there is no visible difference on the script! In these cases you will need to dig deeper to find out why those columns are marked as different. To do that expand the table on the top section and then click on the column you are interested on - on the bottom section of the screen you will see the script for that particular column with the different section highlighted. This difference will be visible on the table script as well however it is a lot easier to "spot" on the column script.

Differences on default constraints, check constraints, default bindings, rule bindings, or extended properties can cause the columns to be flagged as different.

Columns are different but are not marked as such! The highlighting of the differences on the script of an object is controlled by the comparison engine so if the comparison engine marks two objects as equal then nothing will be highlighted on the scripts of those objects even though there might be visible differences on the script. Why does this happen? The answer lies in the comparison options - there are a few options related to collation, nullability, replication, identity etc. that affect the column comparison. So if you have for example chosen to ignore the column collation then even though the column on the left might have a different collation from that on the right the columns will be marked as equal and therefore the collation difference on the script will not be highlighted.

Schema Compare Refreshing Results

When you make changes to comparison options after you have compared two schemas the comparison results might no longer reflect your choices. For example, when you compared the databases, the option **Ignore spaces, new lines and tab characters** was ON so let's say a view named VW1 is marked as being equal on both sides. Now you go to options and uncheck the **Ignore spaces, new lines and tab characters** at which point the VW1 is technically different since let's say its definition on the left schema contains a few extra spaces compared to its definition on the right schema. Any time you make such changes after the comparison IDERA Schema Compare for Oracle Database considers the effect of the changes on the comparison results and if necessary prompts you to refresh the results. The refreshing of the results is accomplished by either a partial or a full re-compare of the schemas.

You can also refresh the comparison results on demand by clicking **Refresh Results** on the ribbon, or by clicking **Refresh Comparison Grid** on the action panel on the lower-right hand side of the comparison results window.

Schema Compare Synchronizing Schemas

Propagating database schema changes from one environment to another (often from development to production) safely and efficiently is the most critical functionality that IDERA Schema Compare for Oracle Database provides to software developers and database administrators.

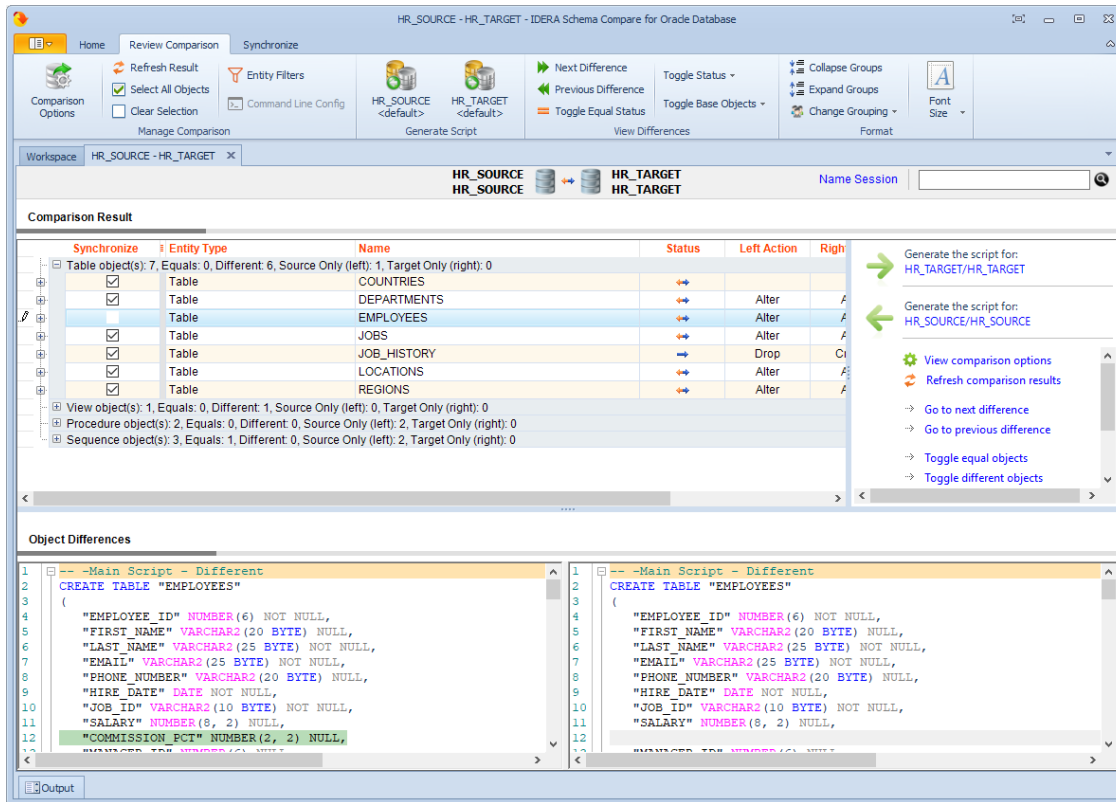
After the comparison of the schemas is complete, you can select the objects you want to include in the synchronization and generate the schema synchronization script. The generated script is displayed on a new tab where you can review and execute it.

Schema Compare Selecting Objects

By default, all the objects that were found to be different during the comparison of the two schemas are marked for inclusion in the synchronization script. However, you can choose to exclude or include certain objects from the comparison by checking or clearing the **Synchronize** checkbox next to each object. Selecting or clearing a parent object also affects the dependent objects. For example, if the checkbox for a table is cleared, then all of its primary keys, foreign keys, indexes etc. are also cleared.

Cleared but included

You may be surprised to see included in the synchronization script an object that you explicitly excluded. Why are those objects included? To answer this you need to look under the scripting / execution options and see if the option to **Always include dependencies of user-selected objects** is **ON** (default). When that option is **ON**, an included object will cause all the objects on which it depends to also be included in the synchronization script. For example, if you have a table T1 that contains foreign keys coming from tables T2 and T3 then, even though you may have cleared tables T2 and T3 if T1 is checked then both T2 and T3 are included. When the **Always include dependencies of user-selected objects** option is **OFF**, then your object selection is respected, however, in this case you should be aware that the resulting synchronization script might not execute successfully.



Schema Compare Generating the Synchronization Script

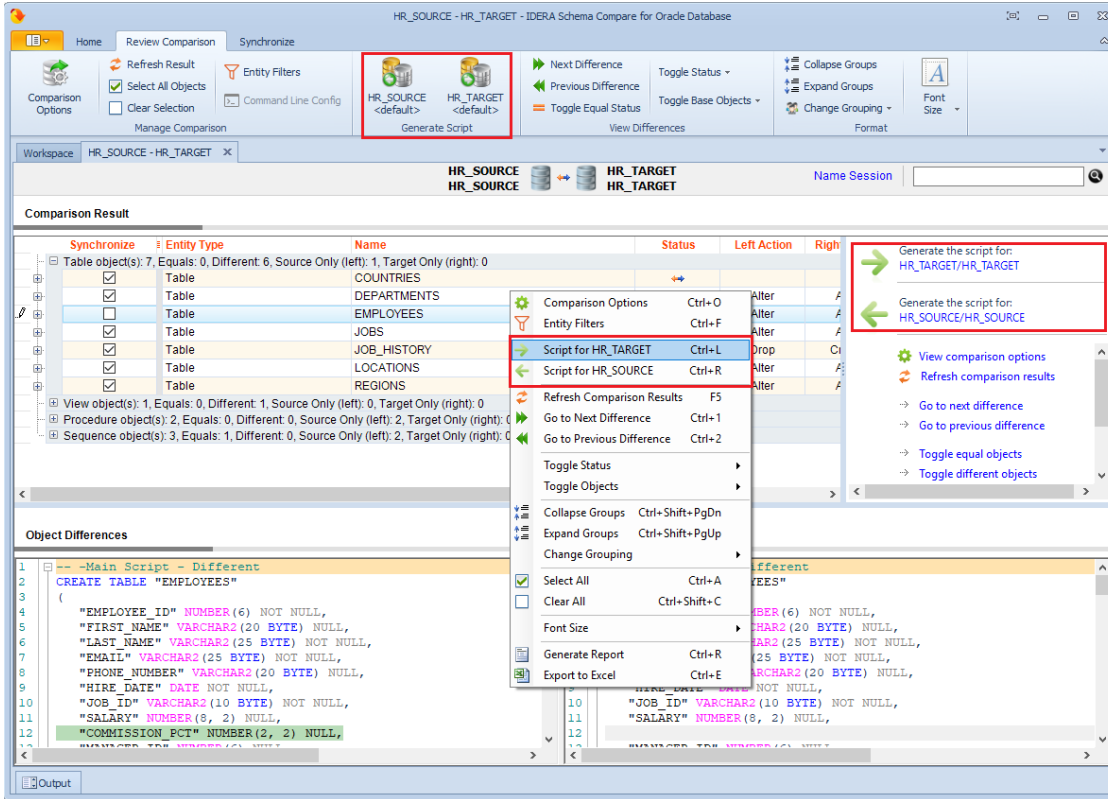
Once the schema comparison operation is finished you are now ready to start the synchronization operation.

By default, all objects that are found to have differences are automatically marked to be included in the synchronization. You can however easily change which objects should or should not be included in the script by checking or clearing the **Synchronization** checkbox (see [Schema Compare Selecting Objects](#)).

Once you make your selections you can generate the synchronization script to either change the database on the right of the comparison to make it the same as that on the left or the other way around. There are three ways to start the script generation:

- **Ribbon.** In the Review Compare tab of the ribbon there are two big buttons in the middle under the grouping **Generate Script**. Each button is labeled with the name of the server plus the name of the database for which the script will be generated.
- **Action Links** on the right panel. (listed under the label **Generate the script for**) Clicking on the Connection/Schema link generates the synchronization script that will make the schema the same as the other one.

- **Context menu.** When you right click on the comparison results grid a context menu will pop up. The context menu allows you to access the comparison options and trigger the generation of the synchronization script in addition of other context relevant actions.



Schema Compare Working with the Synchronization Script

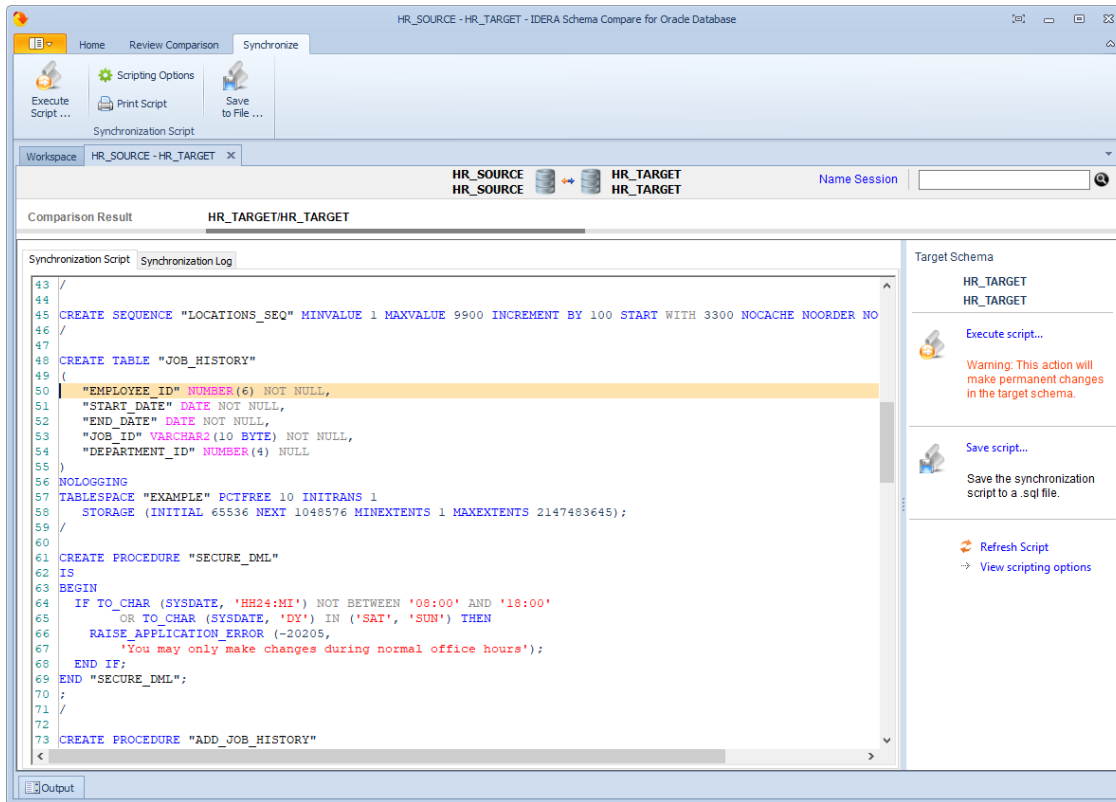
The synchronization script will be displayed on a new window within the comparison sessions tab. The new window is named with the [connection] / [schema] for which the script was generated and it contains two sub-tabs. The Synchronization Script tab contains the actual script whereas the Synchronization Log tab contains a summary of the actions that will be performed on the target database.

Warnings section

It is important that you pay special attention to the warnings displayed at the top section of the synchronization script window. When the synchronization requires dropping objects such as columns or tables that may contain data or in situations when the synchronization has to be performed through an intermediary table, IDERA Schema Compare for Oracle Database will display a warning indicating the potential risk that such action carries.

Both the ribbon and context menu allow you to open the script in SSMS, save the script, copy, print, or jump to a certain line in the script.

Note that if you go back to the comparison results tab and redo the comparison the corresponding synchronization script may be out of sync as the script is not automatically re-generated. To refresh the script, click **generate the script...** on the Comparison Results tab.



Schema Compare Executing Synchronization Script

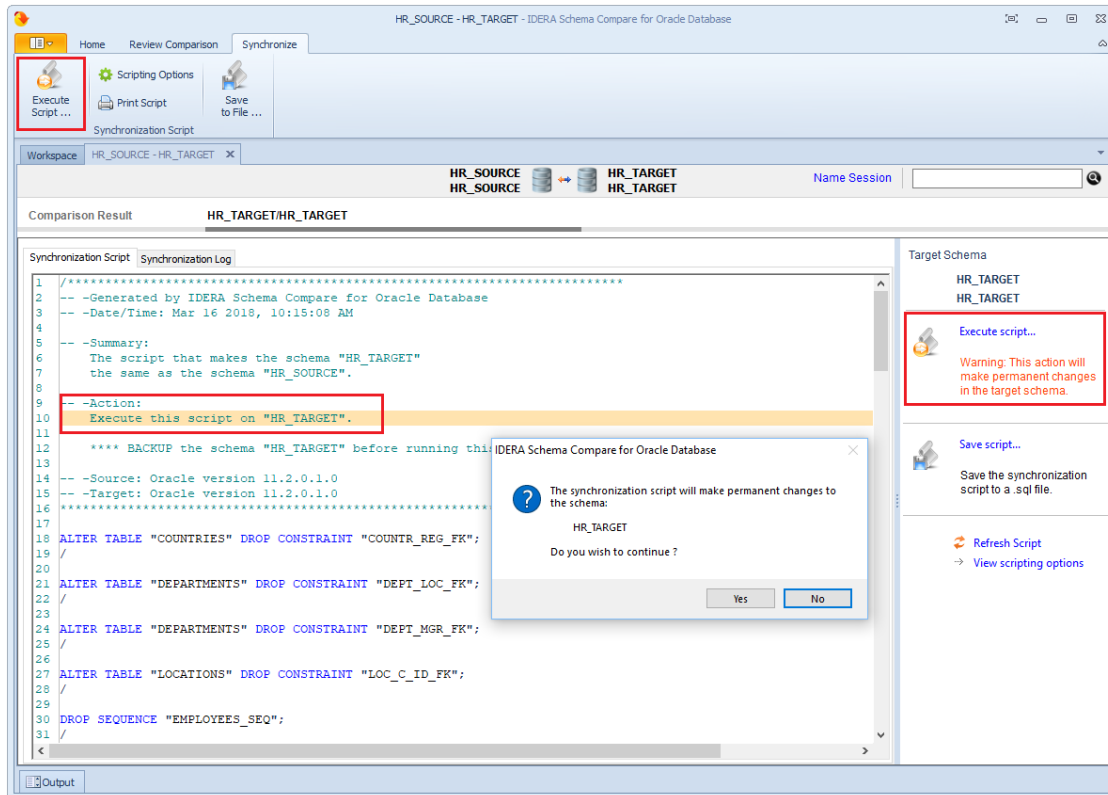
After reviewing the synchronization script and making sure that the script does what you intend, you can execute it.



This action makes permanent changes to the content of the target database. It is common sense and we strongly recommend that before executing the script you:

- Make a full backup of the target schema. Keep in mind that if columns or tables that contain data will be dropped by the synchronization script the only way to reverse the changes will be to restore the data from a full backup of the database.
- Save the script you are about to execute so that you know exactly what was done to the database and when.

You can execute the script directly from IDERA Schema Compare SSMS and execute it from there. Please pay attention to the **Action** section on the header of the synchronization script to make sure that the script will be executed against the correct schema.



Schema Compare Comparison Sessions

Depending on the scenario, the first time you compare two particular databases you might have to spend some time to configure the comparison and synchronization just the way you want it. IDERA Schema Compare for Oracle Database automatically saves all your configuration choices on a "comparison session" so that next time you might need to compare and synchronize those same databases you can do that with one click without having to go through that configuration effort again.

You can also name the comparison sessions to make it easier to identify them. Up to 100 sessions are saved and displayed on the Workspace tab.

The screenshot displays the IDERA Schema Compare for Oracle Database interface. The main window shows a comparison between two schemas: HR_SOURCE and HR_TARGET. A dialog box titled "Name Session" is open, allowing the user to name the session and provide a description. The session name entered is "HR_SOURCE-HR_TARGET" and the description is "Make HR_Target schema the same as HR_Source schema".

The background interface shows a table comparison for "DEPARTMENTS". The table has columns: DEPARTMENT_ID, DEPARTMENT_NAME, MANAGER_ID, and LOCATION_ID. The comparison result shows that the table exists in both schemas, but there are differences in the column definitions. The "Object Differences" section shows the SQL script for creating the table in both schemas, highlighting the differences in column definitions.

Synchronize	Entity Type	Name	Status	Left Action	Right
Table object(s): 7, Equals: 0, Different: 6, Source Only (left): 1, Target Only (right): 0					
<input checked="" type="checkbox"/>	Table	COUNTRIES			
<input checked="" type="checkbox"/>	Table	DEPARTMENTS		Alter	A
<input checked="" type="checkbox"/>	Table	EMPLOYEES			
<input checked="" type="checkbox"/>	Table	JOBS			
<input checked="" type="checkbox"/>	Table	JOB_HISTORY			
<input checked="" type="checkbox"/>	Table	LOCATIONS			
<input checked="" type="checkbox"/>	Table	REGIONS			
View object(s): 1, Equals: 0, Different: 1, Source Only (left): 0, Target Only (right): 1					
Procedure object(s): 2, Equals: 0, Different: 0, Source Only (left): 2, Target Only (right): 0					
Sequence object(s): 3, Equals: 1, Different: 0, Source Only (left): 2, Target Only (right): 0					

```

1  -- Main Script - Different
2  CREATE TABLE "DEPARTMENTS"
3  (
4    "DEPARTMENT_ID" NUMBER(4) NOT NULL,
5    "DEPARTMENT_NAME" VARCHAR2(30 BYTE) NOT NULL,
6    "MANAGER_ID" NUMBER(6) NULL,
7    "LOCATION_ID" NUMBER(4) NULL
8  )
9  NOLOGGING
10 PCTFREE 0 PCTUSED 0 INITRANS 0
11 STORAGE (INITIAL 0 NEXT 0 MINEXTENTS 0 MAXEXTENTS 0 PCTINCREASE 0)
12 /
    
```

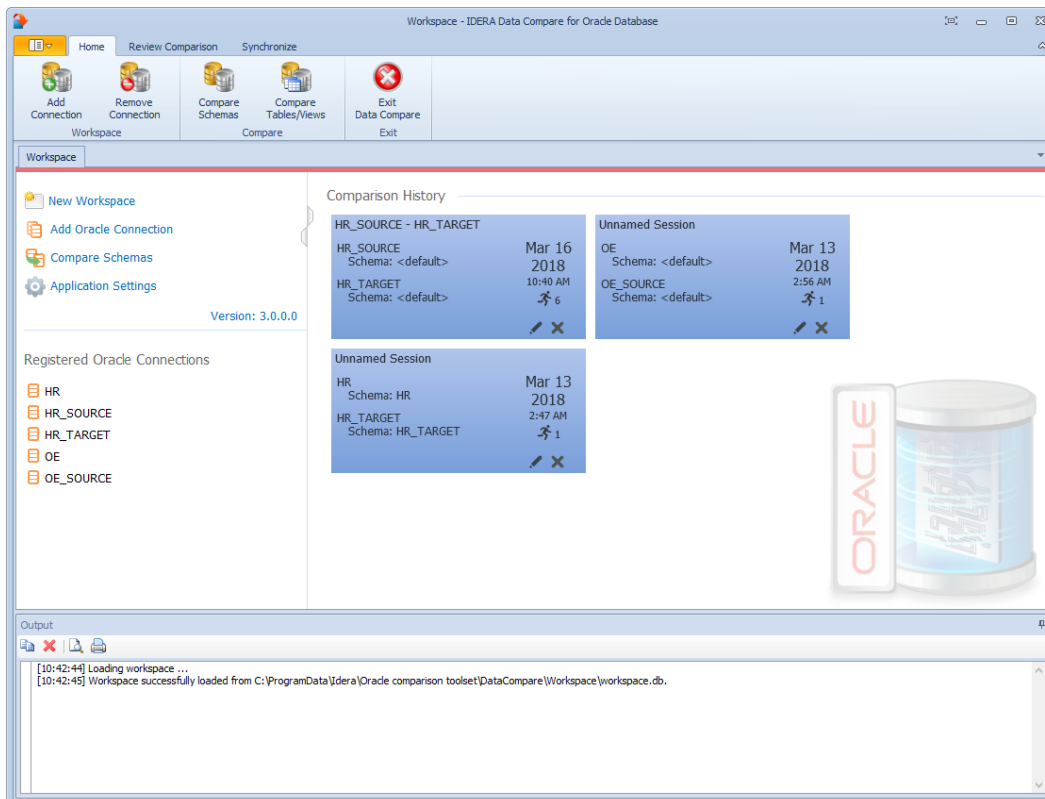
IDERA Data Compare for Oracle Database

Welcome to IDERA Data Compare for Oracle Database, the tool of choice for thousands of DBAs and developers around the globe.

Data Compare Application window

IDERA Data Compare for Oracle Database divides the main application window into three areas:

1. **The ribbon.** The context sensitive ribbon provides all the command buttons and icons you need to do your job quickly. The ribbon is organized in three distinct tabs each of which corresponding to the main steps of the database comparison and synchronization process and "housing" the relevant command buttons.
2. **Main panel.** The main panel contains a permanent tab called the **Workspace** and a tab for each active comparison session.
3. **Output window.** The output window provides a running log of the tasks being performed in IDERA Data Compare for Oracle Database.

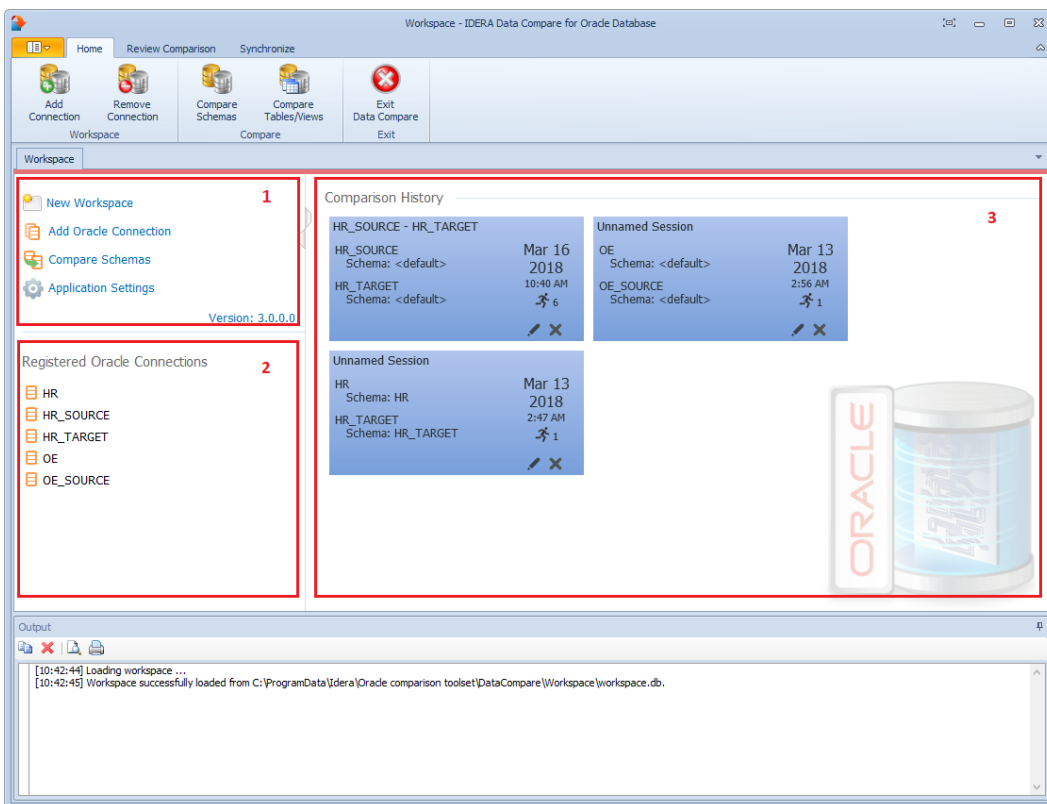


Data Compare Workspace

IDERA Data Compare for Oracle Database workspace is the storage and interface for saving and managing Oracle connections, credentials, comparison sessions, and other UI options.

The Workspace window is divided into the following sections:

- **Workspace links.** This section provides links that allow you to create a new workspace, add an Oracle connection, compare schemas and work with the application settings. The **New Workspace** link resets the current workspace by deleting the entire comparison history and all registered connections.
- **Registered Oracle Connections.** Contains the list of Oracle connections added to the workspace. If you mouse over a connection, you will see action links: **Edit** allows you to change the connection settings; and **Remove** un-registers a connection from the workspace.
- **Comparison History.** Contains the comparison sessions up to the maximum number of sessions specified in the application settings. To launch a session, simply click it.



Data Compare Adding Oracle Connections

Before comparing Oracle schemas, you need to add Oracle connections to the workspace. The IDERA Data Compare Add Connection functionality can be accessed from the ribbon (the Home tab) or from the **Add Oracle Connection** action link at the top left section of the Workspace tab.

On the Add Connection window, you can specify:

- **Connection Name.** A descriptive name for the Oracle connection.
- **Authentication type.** Indicates whether Oracle authentication or operating system authentication will be used. If Oracle authentication is specified, an username and password is required as well.
- **Role.** Indicates the set of privileges associated with the connection. You can choose the default role, SYSDBA or SYSOPER.

In the Basic Settings tab, you can choose the connection type and specify settings required by the connection. There are four connection type you can choose from:

- **Basic Connection.** Requires the name or IP of the host system for Oracle database, the port number and the service name.
- **TNS Identifier.** Requires a connect identifier that exist in the Oracle tnsnames.ora. You cannot use this type of connection to establish a connection to a remote Oracle database. The file **tnsnames.ora** must exist on the local system.

- **Easy Connect.** Requires an easy-connect string for the Oracle database. The format of the easy-connect string is `username/password@[//]host[:port][/]service_name`. Because username and password are already included in the easy-connect string, the credential specified at the top section of the Add Connection form is ignored when the easy-connect option is selected.

- **Connect Descriptor.** Requires a connect descriptor in the format:

```
(DESCRIPTION =
  (ADDRESS_LIST =
    (ADDRESS =
      (PROTOCOL = TCP)(Host = <hostname>)(Port = <port>)))
  (CONNECT_DATA = (SERVICE_NAME = <sid>))
)
```

In the Advanced Settings tab you can specify:

- **Connection Timeout.** A timeout (in seconds) for the Oracle connection. The default value is 30 seconds.
- **Query Timeout.** A timeout value for Oracle queries (command).
- **Pooled.** A flag that indicates whether the Oracle connection should be pooled.

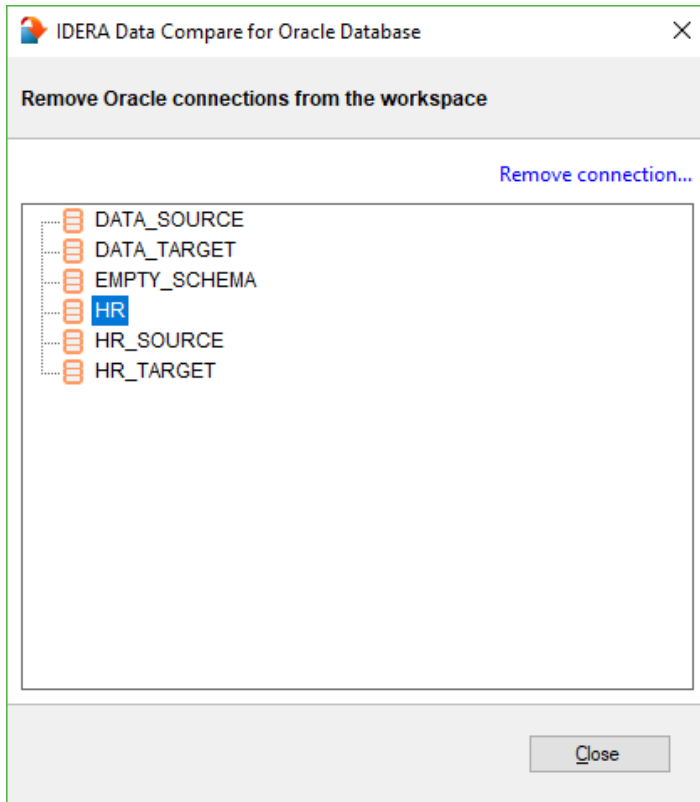
The **Test Connection** link allows you to verify whether the settings are correct and the connection to Oracle database can be established. You can add Oracle connection to the workspace by clicking **OK**.

By default, all Oracle connections use the advanced settings specified in the Application Settings.

Data Compare Removing Oracle Connections

Using IDERA Data Compare for Oracle Database, there are two ways to remove an Oracle connection from the workspace:

- Move the mouse over the name of the connection you want to remove under the Registered Oracle Connections on the left panel of the Workspace. The connection name is highlighted and a **Remove** link is available.
- Click **Remove Connections** on the Home tab of the IDERA Data Compare ribbon. A remove Oracle connections window appears showing all the registered connections. Select the connection you want to remove, and then click **Remove Selected Connection** at the top right corner.



Data Compare Application Settings

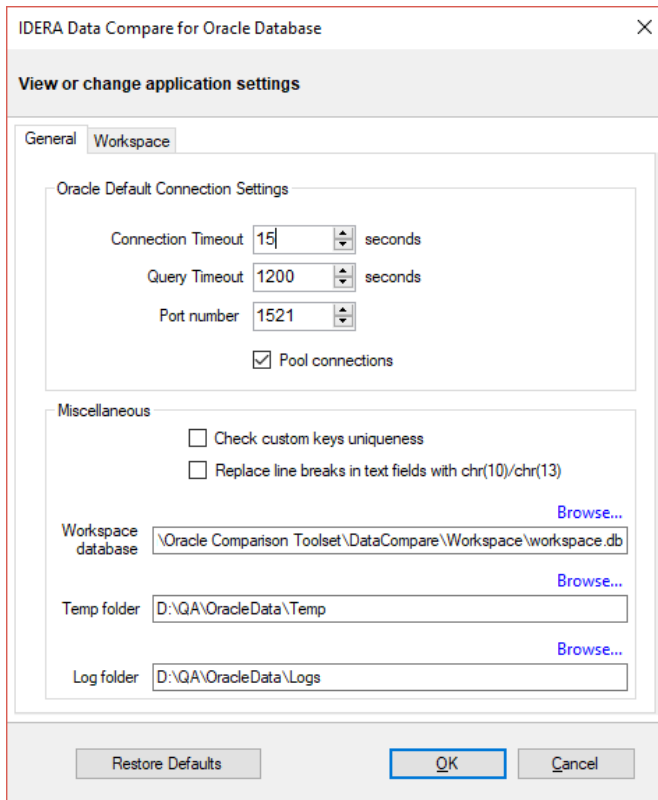
Application settings allow you to tweak the behavior of the IDERA Data Compare for Oracle Database and can be accessed from the drop down menu of the top left corner product icon or from the Application Settings link at the top left corner on the Workspace tab.

Under the General tab, you will find:

- **Oracle Connection.** If you are reading this you are already familiar with the connection settings. Under normal circumstances you do not need to change any of these settings.
- **Check custom keys uniqueness.** Before comparing the data, IDERA Data Compare for Oracle Database will pair the rows based on a unique key that is automatically determined. However, you can choose to pair the rows based on some custom key that you define either because the tables in question do not have a predefined unique key or because you wish to use a different key. In this case there is an option to validate the chosen key. By default the validation simply checks for data type compatibility. If you wish to ensure that the chosen key is unique you can check this option. Please note however that depending on the size of those tables the validation for uniqueness may take from a few seconds to minutes to complete.
- **Replace line breaks in text fields with chr(10)/chr(13).** Sanitizes the line breaks in the text fields of the data synchronization script. Some third-party tools do not accept line breaks in text data, so the

purpose of this option is to generate a synchronization script that runs successfully outside IDERA Data Compare for Oracle Database. This option is not required to be set when the synchronization script is executed within data compare.

- **Workspace database.** The path and the name of the workspace database file. You can change it by clicking **Browse** and then selecting a custom file.
- **Temp folder.** Indicates the location where Data Compare stores temp files that are created during the comparison. You can change the location by clicking **Browse** and then selecting a custom path.
- **Log folder.** Location where data compare stores the files that are created during the comparison. you can change the location by clicking **Browse** and then selecting a custom path.

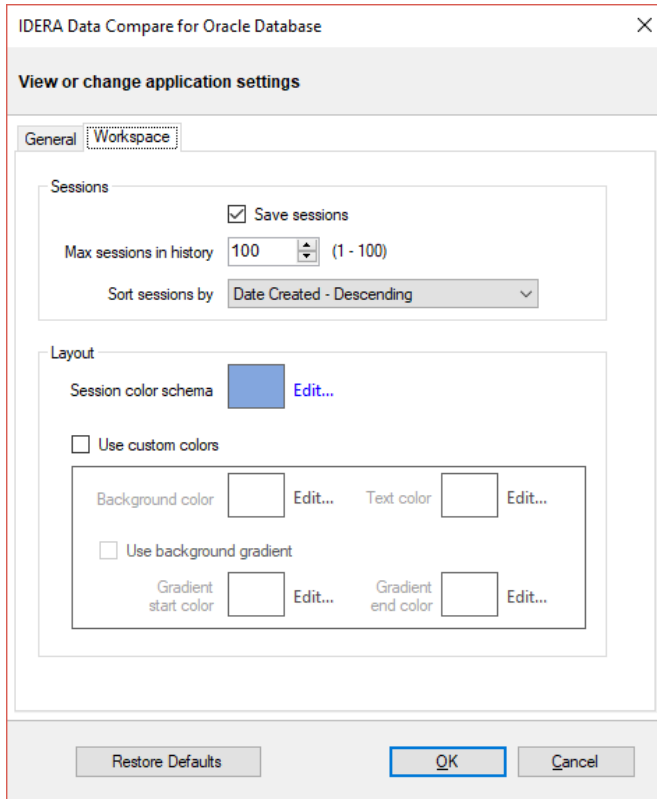


Under the Workspace tab you will find:

- **Sessions.** Contains settings related to comparison sessions:
 - **Save sessions.** By default it is checked. It instructs the IDERA Data Compare to store comparison sessions for future use. A stored session contains all the necessary information to allow you to repeat the comparison with one click. Stored sessions will appear on the main panel of the Workspace tab.
 - **Max sessions kept in history.** By default IDERA Data Compare will store the last 100 comparison sessions. Although there is no noticeable performance difference on application launch related to the number of sessions stored you may choose to set this parameter to a lower number. Allowable values are 1 to 100.
 - **Sort sessions by.** Change the order that stored sessions appear on the workspace tab.

- **Layout.** Allows you to change the color schema of a session. You can choose from built-in colors or pick custom ones.

To revert all settings back to their default values, click **Restore Defaults**.



Data Compare Comparing Data

Comparing the data in two schemas is part of the core functionality of IDERA Data Compare for Oracle Database.

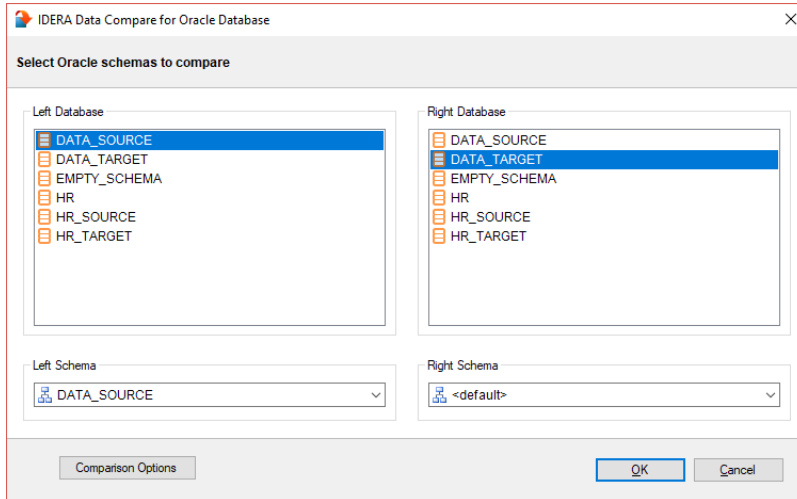
The databases you wish to compare (or tables within them) can be virtually anywhere as long as you have access to them you can compare and synchronize them. Furthermore, the databases do not have to be of the same version or flavor of Oracle.

In the case of cross version comparison, data compare will map the data types that can be mapped and ignore the columns that are of a type that cannot be mapped.

Data Compare Comparing Oracle Schemas

To start a comparison between two database schemas, click on the IDERA Data Compare for Oracle Database **Compare Schemas** button on the Home tab of the ribbon. The Oracle Schema Selection window that appears allows you to choose the Oracle connections and the Oracle schema on each connection. If you do not specify a schema, the default schema associated with the Oracle connection's user is used.

From this window, you can also access the [Comparison Options](#) that are applied during the comparison.



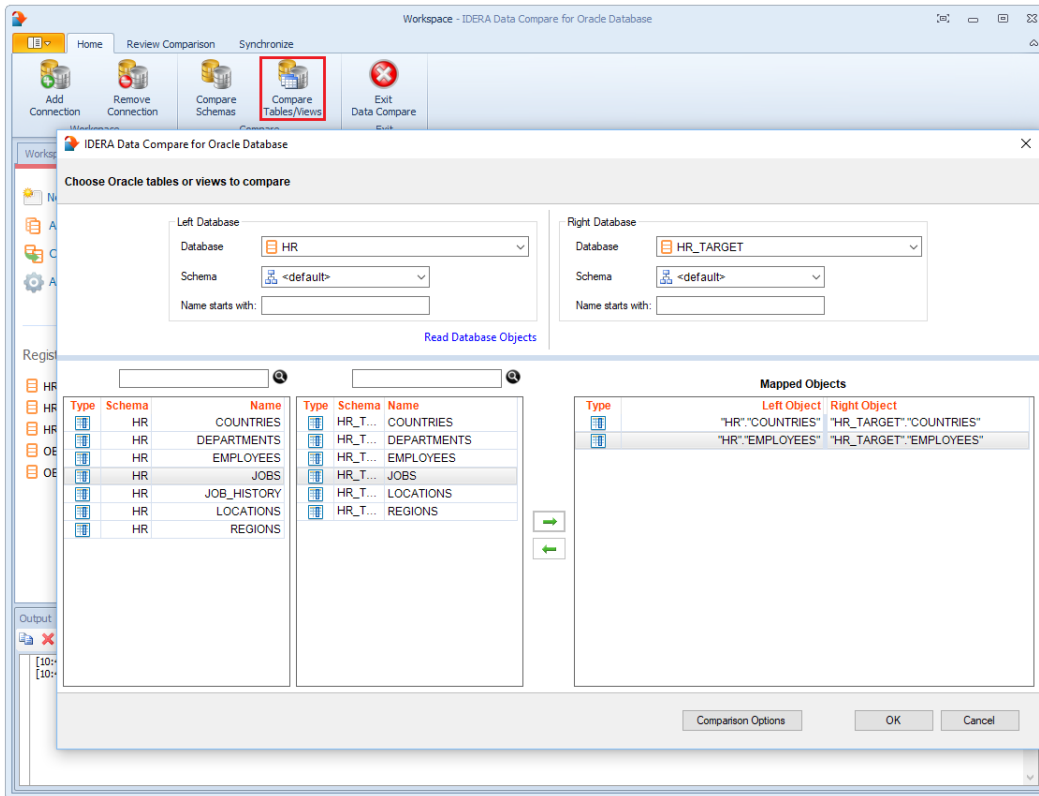
Data Compare Mapping Tables Manually

Data compare, by default, maps Oracle tables on two schema by name. To map them manually, click **Compare Tables/Views** on the "Home" tab of the ribbon. On the Object Compare window that appears you can first select the Oracle connections and schemas on which the objects reside. When there is a large number of objects you may wish to specify a pattern for the name of the objects you are looking for (ex. all tables that start with "a"). That makes it easier to find those objects.

IDERA Data Compare for Oracle Database, in addition to tables, can compare views. The Compare and Synchronize Views option determines whether views should be included in the comparison and synchronization. However, use this option with caution. There are limitations that Oracle imposes on views and not all views are updatable.

Once you make your selections, click **Read Database Objects**. Data compare will read objects on each schema and populate the two left bottom windows with the list of tables / views available.

Now you can map those objects one at a time irrespective of the object names. Select an object on the left, select an object on the right, and then click the left to right "arrow" button to map those two objects.

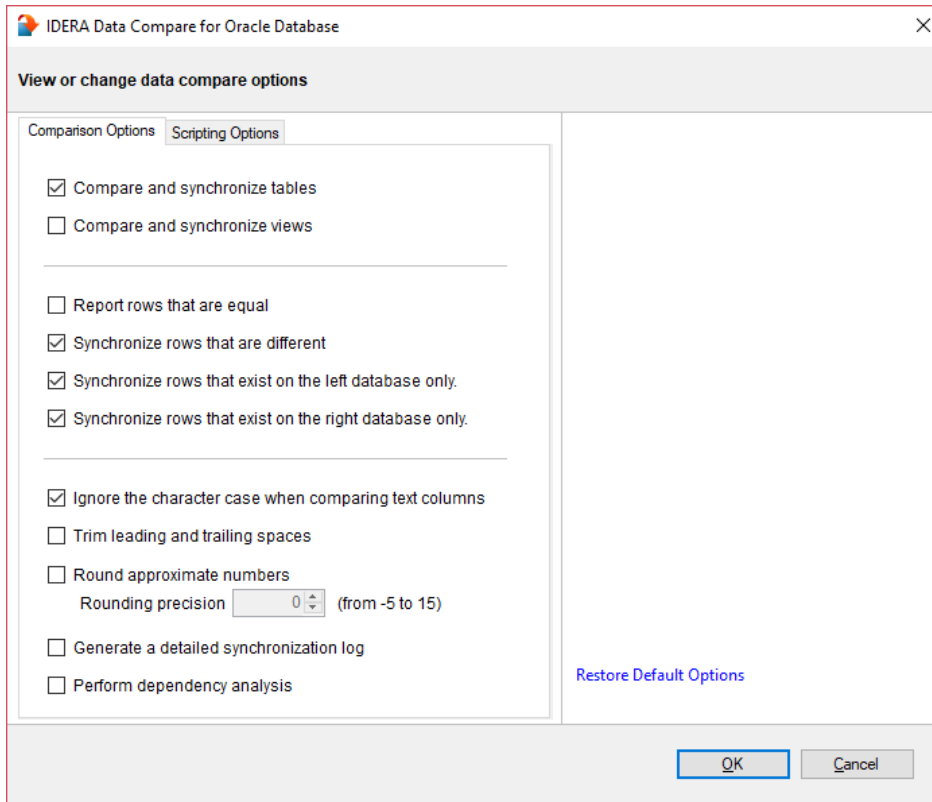


i The comparison options will affect the ability to compare the objects. If, for example, you cleared the **Compare and synchronize tables** option, then while you can still see and map the tables they will not be compared. The same holds true for views which by default are not included in the comparison.

Data Compare Comparison Options

IDERA Data Compare for Oracle Database exposes a series of options that allow you to customize the way the comparison is performed. On mouse over a description of the highlighted option is displayed on the right panel of the options window.

You can access the Data Compare options from the Compare Databases and Compare Tables dialog windows before the object mapping operation. After the objects (tables and indexed views) have been mapped you can access the Comparison Options from the Review Comparison tab of the ribbon as well as from the action links on the right panel of the Mapped Objects tab.

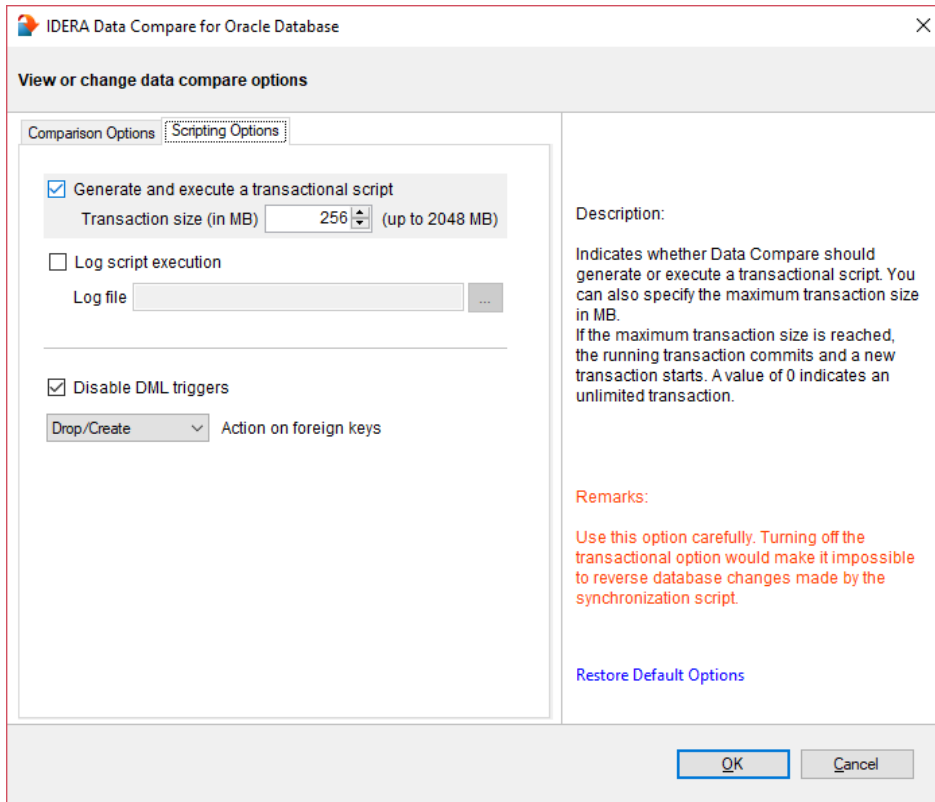


Data Compare Scripting Options

Scripting Options are listed on the second tab of the IDERA Data Compare Options window. Those options affect the way the synchronization script is generated but do not affect the comparison operation.

- Transaction size.** Determines the size of the script that will be wrapped in a single transaction. Depending on the size of the databases/tables being compared the synchronization script may get to be extremely large. Wrapping such large scripts in a single transaction may not be possible due to transaction size limitations, and may not be acceptable for performance and other reasons. So, it is necessary to split the script into smaller "chunks" and wrap each of those "chunks" in individual transactions. If the synchronization process is canceled or fails due to errors, only the current open transaction is rolled back. Changes committed prior to that transaction are permanent and cannot be reversed. To have an unlimited transaction size specify 0 as the max value. Having an unlimited transaction size ensures that all changes in the database are rolled back if an error occurs, but might be rejected by Oracle.

i Finding the right transaction size depends on many factors - please adjust this option carefully to suit your specific environment.



Data Compare Preparing the Comparison

Before comparing the content of two Oracle schemas, data compare performs a mapping of schema objects, i.e. tables and, optionally, views. By default, a table on one schema is mapped to a table on the other schema that has the same name. Depending on the comparison options, views and can also be mapped with each other. If you choose to compare specific tables or views, then data compare honors the manual mapping, compares only the selected objects and ignores the rest of the schema.

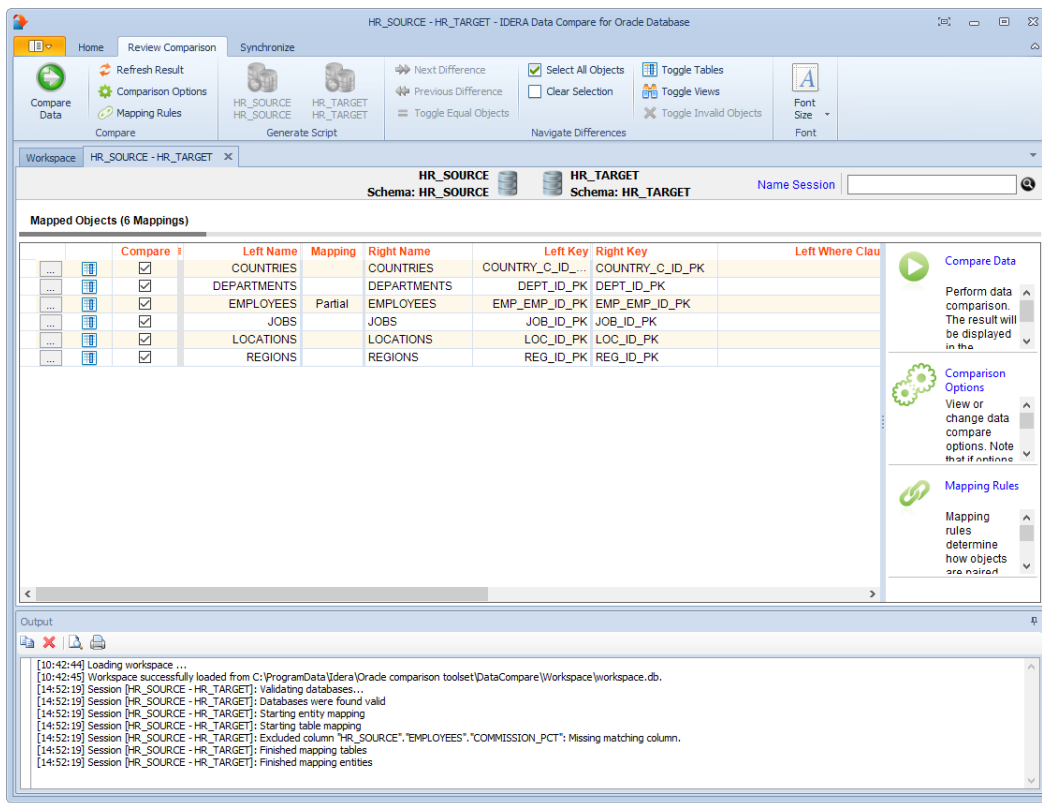
Once the object are mapped, data compare analyzes columns and chooses a suitable unique key that will be used to compare the data rows.

During this preparation phase you will be able to customize the object mapping, include and exclude certain tables and certain columns within those tables, choose comparison keys and define data filters to compare only subsets of rows.

Data Compare Mapped Objects

The results of the schema analyses and the automatic mapping operation are displayed on the Mapped Objects tab. For each pair of mapped objects the following information is displayed:

- **Object Type.** The object can be a table or a view. The type is indicated through the small icon in the second column of the grid.
- **Compare.** A checkbox indicating whether the pair should be included in the comparison. By default, all mapped pairs are included in the comparison. You can quickly select / deselect all objects using the **Select All Objects** and **Clear Selection** commands in the ribbon.
- **Object name.** For both sides. In a default mapping the **Left Name** would be identical to **Right Name**.
- **Mapping Status.** A blank value indicates that the objects have been fully mapped, that is, there was a complete match of columns. A **Partial** value indicates that some columns were not matched and will not be included in the comparison.
- **Left/Right Keys.** Those are the unique keys that have been selected by Data Compare to be used in the mapping of the rows during the data comparison. If those two fields are empty for a given pair that means that Data Compare was not able to identify a unique key that could be used. If you do not choose a custom key for this pair then those objects will not be included in the comparison.

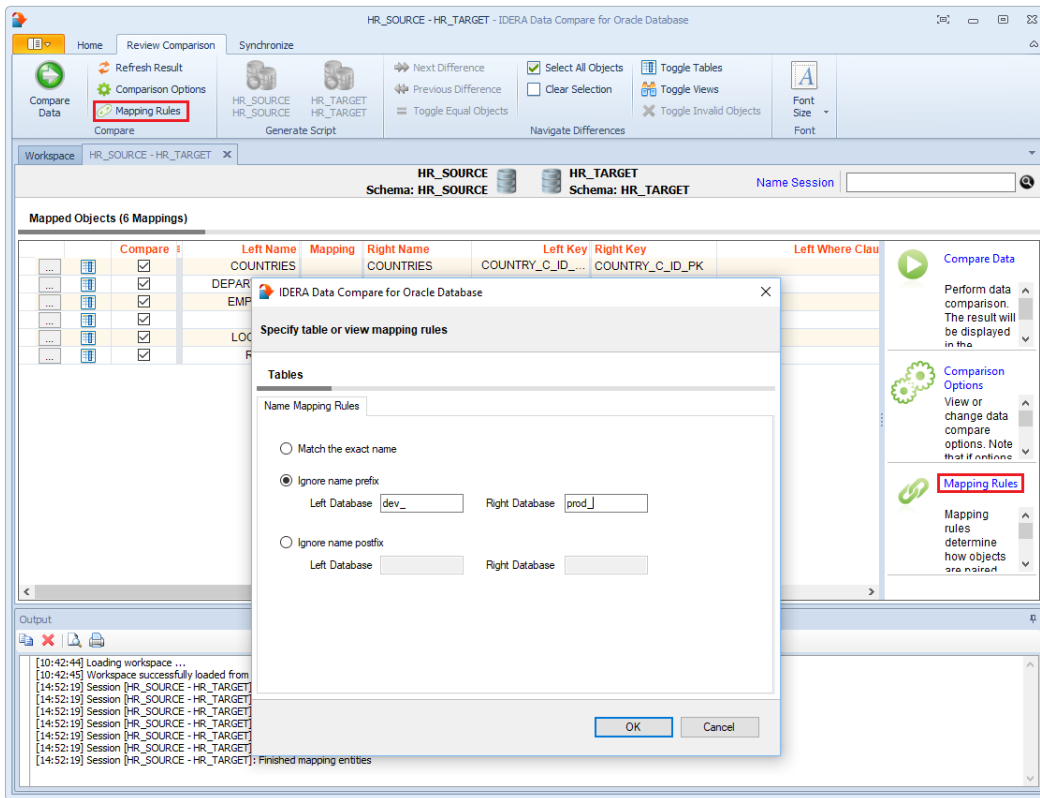


Data Compare Mapping Rules

In most cases the default mapping mechanism that pairs the objects based on the selected schemas and name is what is needed. However, there are often scenarios in which tables on the development server for example may have names

prefixed with dev_ whereas the same tables in the production server do not have that prefix. In such cases you may want to define certain mapping rules that would allow you to pair objects that would otherwise not be paired together by Idera Data Compare for Oracle Database.

The Mapping Rules can be accessed from the ribbon or from the action links on the right panel of the comparison tab. By default data compare performs an exact match on the object names. However, you can choose to ignore certain prefix or postfix. For example, if you ignore the "dev_" prefix on the left schema and the "prod_" prefix on the right schema, a table named dev_employees on the left schema will be mapped to a table named prod_employees on the right schema.

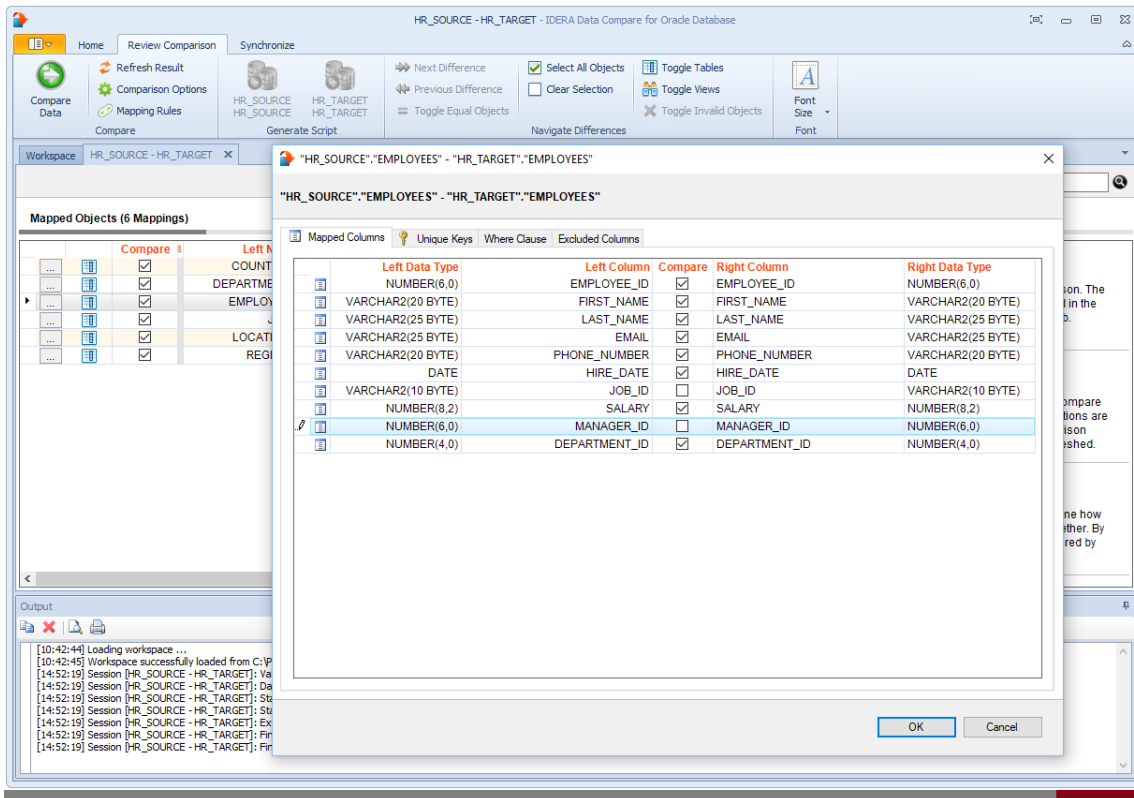


Data Compare Excluding Tables and Columns

The first step in customizing your comparison after the objects are paired is to choose which objects and columns you wish to exclude from the comparison. The options available in Idera Data Compare for Oracle Database include:

- **Excluding Objects.** By default all objects (tables and views) are included in the comparison. You can exclude them by un-checking or clearing the **Compare** checkbox in the **Mapped Objects** grid. You can also use **Select all objects** to include all objects or use **Clear selection** to exclude all objects in one click. Both options are available in the Data Compare ribbon.

- **Excluding Columns.** Click the details button (the first column of the **Mapped Objects** grid) to view and customize a specific object. To exclude columns simply uncheck (clear) the **Compare** checkbox.



Data Compare Comparison Keys

The IDERA Data Compare Unique Keys tab is divided in two main sections. The top section contains two boxes with the list of unique keys that are potential candidates to be used as the comparison key for each table in the pair. Initially the boxes contain only the predefined unique keys that Data Compare found, but you can add other unique keys that can be a combination of any and all the columns on the given table.

The bottom section contains a box with the pair of unique keys that have been selected as the key to be used for the data comparison operation.

Data Compare picks the comparison key in the following order (whichever is found first):

1. User-defined keys. User-defined keys (or custom keys) takes precedence over the built-in keys and indexes.
2. Primary Key
3. Unique Constraints

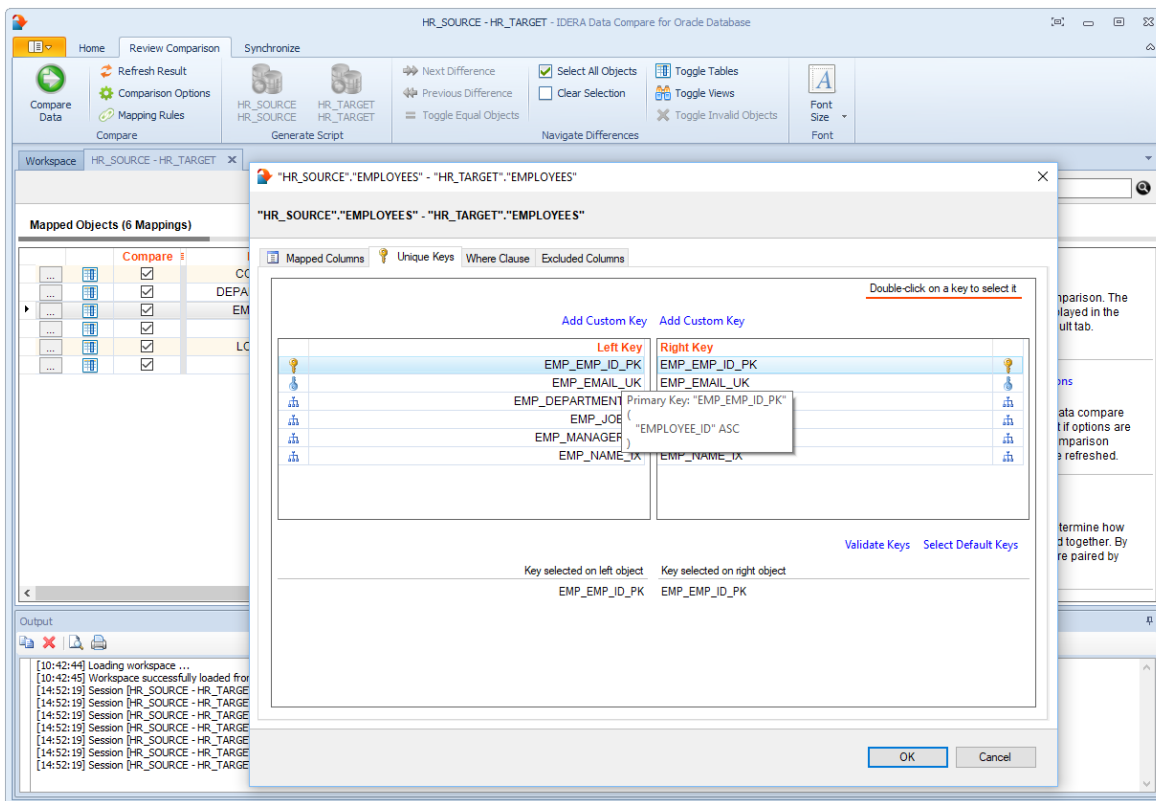
4. Unique Indexes (Indexes that are not unique are not considered.)

When custom keys are selected, you should validate the keys by clicking **Validate**. By default, the keys are validated only for data-type compatibility. If you wish to validate your custom defined keys for uniqueness, then you should check that option in the **Application Settings**.

User-defined key validation is done as follows:

- Custom keys must have columns with the same name.
- Columns in the custom keys must have the same data type. Length, scale, precision, nullability, and other column properties are not considered.
- The uniqueness of the key is checked only if the **Check custom key uniqueness** option in the **Application Settings** is checked.

To select a key for data comparison, simply double click the appropriate key.



Data Compare Where Clauses

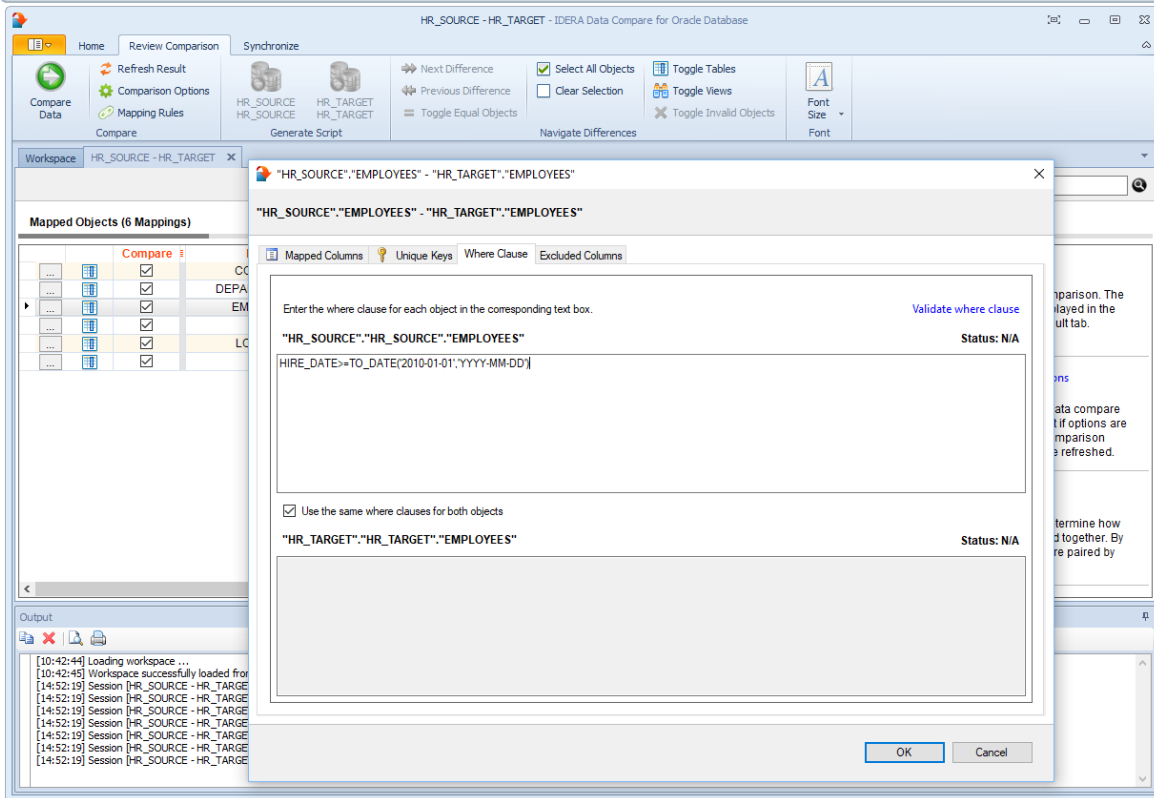
For large tables the comparison operation may take time since all the rows from both sides will have to be read and compared. So, if you know for example that out of 10 million rows, that 9 million of them are equal, it would be very

inefficient to compare those equal rows. Furthermore, regardless of how many rows may be different you might need to do a quick comparison of just a subset of rows, for example, only rows that were modified today.

IDERA Data Compare for Oracle Database allows you to set where clauses for each object being compared so that you can read and compare only the rows you need to. When defining a where clause by default the same filter will be applied to both tables. If you wish to define different filters for each of the tables you can un-check or clear the **Use the same where clause** checkbox.

i Do not include the "where" keyword in the where clause. Simply enter the condition in the same way you would write it in any Oracle tools.

i Validate the where clause before comparing the data. This ensures that the clause is defined correctly.

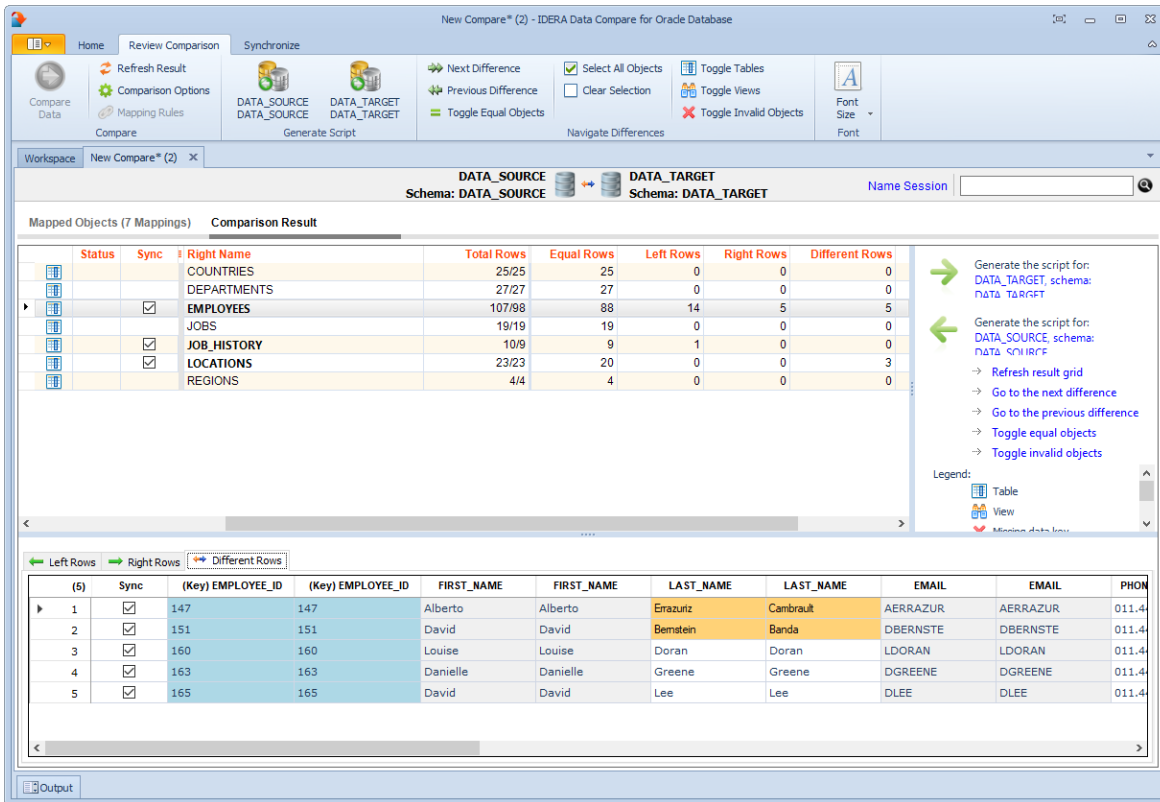


Data Compare Comparing the Data

After completing the preparation phase of the comparison you are ready to compare the data in all those mapped objects (tables and views). To start the comparison, click either **Compare Data** on the ribbon (under the Review Comparison tab) or click **Compare Data** on the right panel of the Mapped Objects window. A progress window displays the start time, elapsed time, and the names of the objects being compared. If you cancel the operation, data compare will display partial results.

The comparison result window is divided into the following sections:

1. **Comparison summary.** Displays the tables/views that have been compared and the row count as follows:
 - **Equal Rows.** Rows that are equal.
 - **Left Rows.** Rows that exist only on the left table/view.
 - **Right Rows.** Rows that exist only on the right table/view.
 - **Different Rows.** Rows that exist on both sides and are different.
Objects with data differences are displayed in bold and have a checkbox, that either includes or excludes them from the synchronization script. An object with no data key is considered "invalid." Invalid objects can't be compared.
2. **Comparison details.** When you click on a pair of objects, depending on the data differences between them, one to three tabs will appear in the bottom section of the comparison window: **Left Rows**, **Right Rows**, and **Different Rows**. The grid that contains different rows, shows the paired columns from both objects side-by-side, so that you can easily see data differences. Right click on a row to view more the row data in more detail.
Notes on details grid:
 - Columns in the light-blue color are the columns that participate in the data key. The “key” label appears in the column header as well.
 - Cells in orange color represent data values that are different.
 - Rows that are equal, by default, are not shown in the row grid. To view equal rows, check the comparison option **Report rows that are equal**.
3. **Generate synchronization script.** Those links will trigger the generation of the synchronization script for one direction or the other.
4. **Comparison action links.** The action items provided in this section are also available on the Review Comparison tab of the ribbon. Clicking **Refresh Results** triggers a re-compare of the data.
5. **Legend.** Provides a list of symbols that are used in the results grid.



Data Compare Synchronizing

Identifying the data differences between two databases helps you as a DBA/Developer in many ways but often your job does not end there. You need to migrate those changes from one database to another. IDERA Data Compare allows you to do just that. After comparing the contents of two databases you can use it to generate the synchronization script that will update the target database (or just the selected tables/rows) to make it the same as the source database.

The data synchronization operation consists of three key actions:

- **Update.** All rows that exist in both sides will be updated on the target database to match the source.
- **Insert.** All rows that exist only on the source database but not on the target will be inserted on the target database.
- **Delete.** All the rows that exist only on the target database but not on the source will be deleted from the target database.

In case you do not wish to delete the rows that only exist on the target database you can change the comparison options so that those rows are not included in the synchronization.

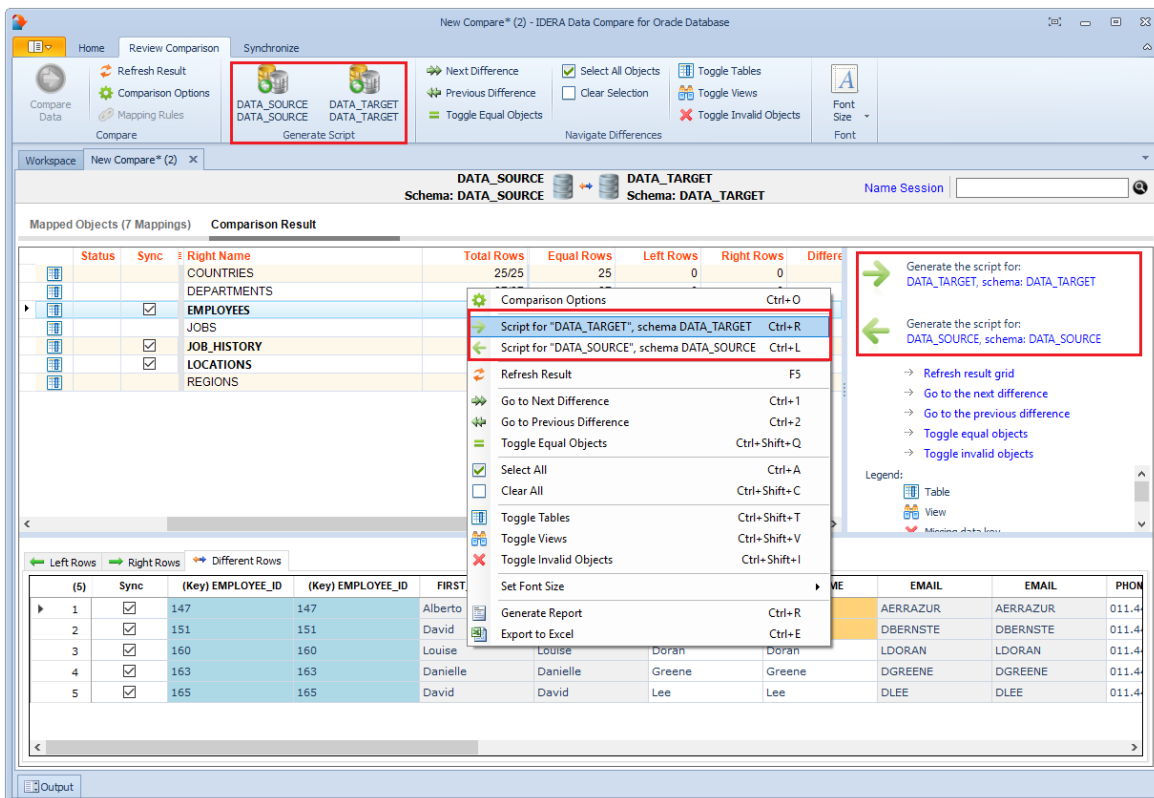
Data Compare Generating Synchronization Script

Once the data comparison operation is finished you are now ready to start the data synchronization operation.

By default, all tables that are found to have differences and all the different rows in those tables are automatically marked to be included in the synchronization. You can however easily change which tables, and which rows in those tables, should or should not be included in the script by checking or un-checking (clearing) the **Sync** checkbox.


Once you have made your selections you can generate the synchronization script to either change the database on the right of the comparison to make it the same as that on the left or the other way around. There are three ways to start the script generation:

- **Ribbon.** In the Review Comparison tab of the ribbon there are two big buttons in the middle under the grouping **Generate Differences Script**. Each button is labeled with the name of the server and name of the database for which the script will be generated.
- **Action Links on the right panel.** Are listed under the label **Generate the script for**. Clicking on the [connection].[schema] link generates the synchronization script that will make that database the same as the other one.
- **Context menu.** When you right click on the comparison results grid a context menu will pop up. The context menu allows you to access the comparison options and trigger the generation of the synchronization script in addition of other context relevant actions.



Data Compare Working with the Script

the data synchronization script is displayed on a new window within the comparison tab. the new window is named with the [Oracle connection], [schema] for which the script was generated.

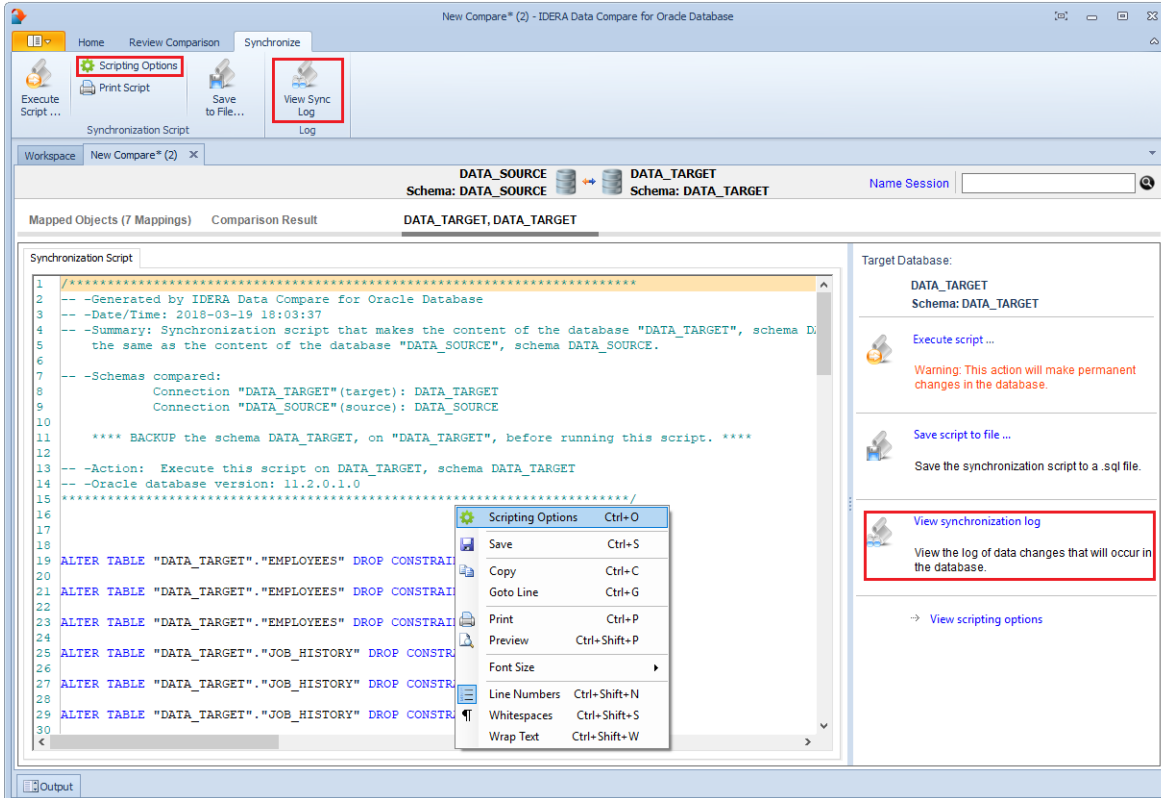
 If the data synchronization script exceeds 32MB, Data Compare displays only the first 32MB-size block of the script in order to preserve the memory. If you want to view the whole script, you can save it to a file, and then open the file using a text editor.

A handy context menu allows you to save, copy, print, and add line numbers to the script as well as jump to a certain line in the script.

Here you can review the script to ensure that it is doing what you intend it to do. You may want to start the review from the Synchronization Log so that you can quickly identify the areas of the script that may require a closer look. You can access the synchronization log from the ribbon by clicking **View Sync Log** or from the action links on the right panel. The synchronization log contains a summary of the actions that you want to perform on the target database.

Refreshing the comparison and regenerating the script

While reviewing the script, you may discover that you should have done something different during the preparation phase before running the compare. IDERA Data Compare for Oracle Database allows you to change the options at any point during the process, and the simply click **Refresh Result** on the ribbon to redo the comparison without having to go through the whole exercise of selecting columns and keys etc. Note that this refresh action does not affect the previously generated synchronization script. To refresh the script window, simply generate the script again.

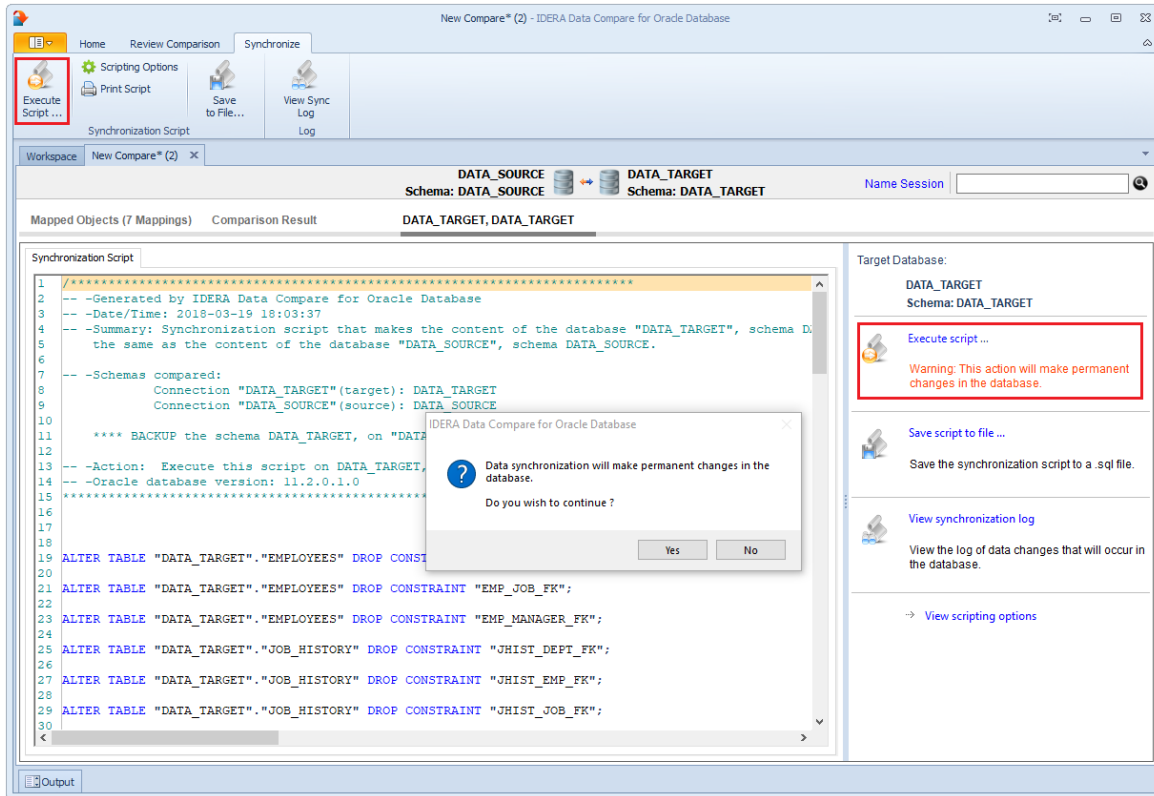


i If the synchronization script contains LOB fields (CLOB, NCLOB or BLOB) whose size exceeds the maximum inline size supported by Oracle (4000 bytes), IDERA Data Compare for Oracle Database adds some additional steps to the core synchronization script to overcome this limitation. An Oracle package is created before the data script begins the execution and it is dropped after the execution. The package allows the LOB fields to be updated in chunk of approximately 4000 bytes in size.

Data Compare Executing Synchronization Script

Once you have reviewed and you are comfortable with the synchronization script you can execute it on the target database.

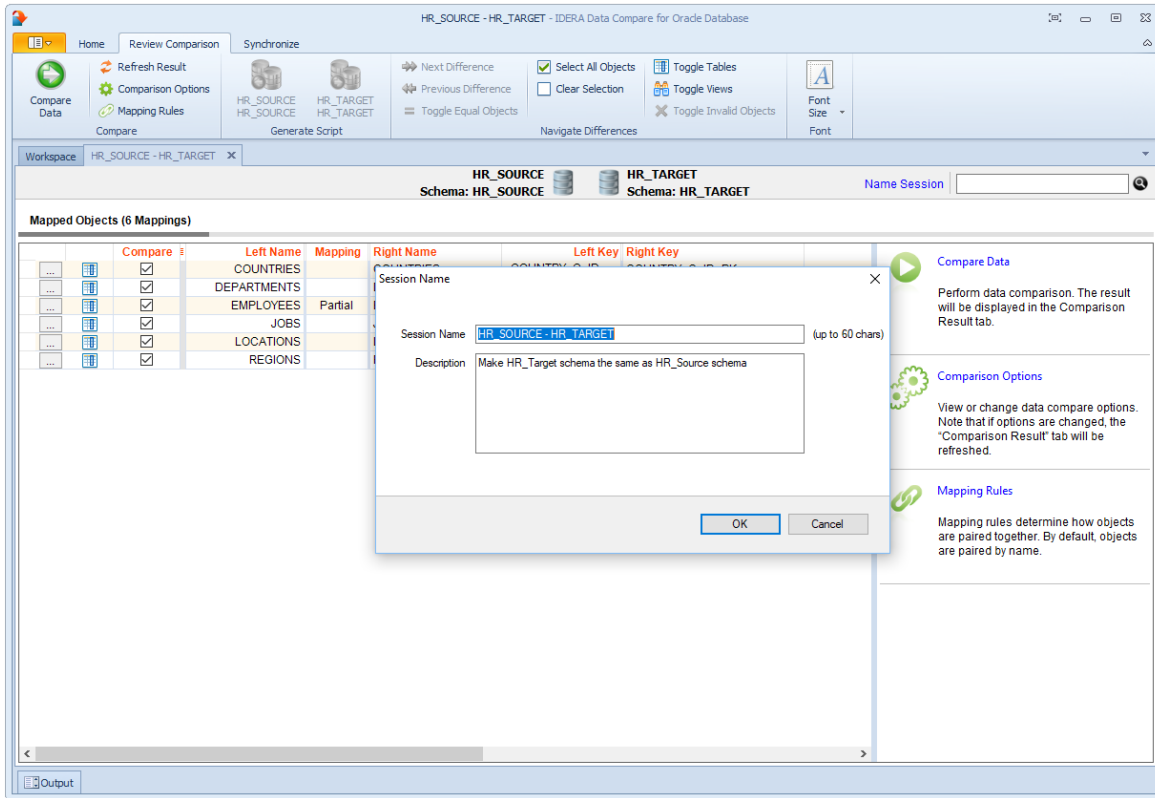
⚠ This action makes permanent changes to the content of the target schema. We strongly recommend that you back up the schema before executing the script.



Data Compare Comparison Sessions

The preparation phase of the data compare can be a time consuming process if you need to exclude certain objects, do custom mapping, define custom comparison keys and set data filters. IDERA Data Compare automatically saves all your preparation work so that you can repeat the same comparison in the future with a single click without having to go through that process again.

You can also name the comparison sessions to make it easier to identify them. Up to 100 sessions are saved and displayed on the Workspace tab.



IDERA Data Compare command line

IDERA Data Compare for Oracle Database command line provides for comparing and synchronizing the content of Oracle databases via the command line.

With the command line you can:


- Perform automated data comparison and synchronization;
- Compare and synchronize database content as part of your setup and deployment solution;
- Compare database content and execute the data synchronization script in batch files;
- Schedule the comparison.

The command line comes with samples for most typical data comparison scenarios. The sample configuration files are located under **\Program Files (x86)\Idera\Oracle Comparison Toolset\Command Line Samples\Data Compare**.

Data Compare Command Line Arguments Overview

The command line supports two categories of options.

- Options that are specified as command line switches. Most of the command line switches have a short and a long name and separate the name from the value by a semicolon.
- Options that are specified via an xml configuration file. We strongly recommend that you set all command line options via the xml configuration file.

 When the option can be specified via the command line and xml file, the command line switch takes precedence.

IDERA Data Compare command line supports the following command line switches:

Option	Description
xmlfile	Performs a data comparison using settings in the specified xmlfile .
/datascript:filename	Specifies a file name for the data synchronization script. We recommend you set this option in the xml configuration file.
/datalog:filename	Specifies a file name for the data synchronization log. We recommend you set this option in the xml configuration file.
/en, /encoding: (UTF7 UTF8 UNICODE ASCII)	Indicates the encoding that should be used for output files. ANSI encoding is used if no encoding is specified.
/t, /logtype:(windowslog)	Indicates the type of error log. You can specify windowslog to log the errors in Windows Event log.
/e, /error:errorlog	Specifies the error log. Depending on the /t option, it could indicate a file or a Windows Event log.

Option	Description
/l, /logfile:filename	Specifies a file for the data compare operational log. We recommend you specify the option in the xml configuration file.
/o, /overwrite	Indicates whether the log files should be overwritten. By default command line appends to output logs.
/q, /quiet	Specifies the "quiet" mode of operation, which suppresses all messages from being displayed on the console.
/d, /direction: (leftasright rightasleft)	Indicates whether the command line should make the right database the same as the left database or vice versa. You can specify leftasright or rightasleft . We recommend you set this option in the xml configuration file.
/s, /sync	Indicates whether the command line should synchronize databases. By default data is only compared but the synchronization script is NOT executed. Use this option carefully since it makes permanent changes in the target database.
/mappingtype: (auto custom)	Indicates the mapping of database tables and views. If auto is specified, objects are mapped based on the default mapping: schema.name. If custom is selected, the pairs of objects must be specified manually. The default mapping type is auto. We recommend you set this option in the xml file.
/v, /validate	Validates the xml configuration file.
/? , /h, /help	Displays the command line options.

Data Compare XML Configuration File Overview

Xml configuration file allows you to specify all data compare options and settings. Even though some settings can be set via command line switches, we recommend you specify them in the xml file.

With the xml configuration file you can:

- Specify databases and login credentials;
- Define the comparison options;
- Customize data compare by excluding objects, defining custom keys and where clauses;
- Specify mapping rules;
- Specify the files for the synchronization script and logs;
- Execute the data synchronization script.

The xml file groups the comparison options into the following sections:

Section	Description
Database <LeftDatabase> <RightDatabase>	Contains the source and target database (or schema).
Data Compare Options <DataCompareOptions>	Contains the data compare options.
Table Mapping Rules <TableMappingRules>	Contains the rules that determine how database tables are mapped.
View Mapping Rules <ViewMappingRules>	Contains the rules that determine how database views are mapped.
Table Pairs <TablePairs>	Contains the pairs of tables that should be mapped manually or pairs that should be excluded.
View Pairs <ViewPairs>	Contains the pairs of views that should be mapped manually or pairs that should be excluded.
Command Line Settings <CommandLineOptions>	Contains additional data compare settings.

A typical xml configuration file has the following structure:

```
<?xml version="1.0" encoding="utf-8" ?>
<DataCompareCmdConfig xmlns="http://www.idera.com/oracle/schemas/datacompare/cmd.xsd">
  <LeftDatabase>
    <ConnectionName>Oracle Schema 1</ConnectionName>
    <ConnectionType>Basic</ConnectionType>
    <Username>HR</Username>
    <Password>*****</Password>
    <Host>host-name-or-ip</Host>
    <Service>oracle-service-name</Service>
    <DatabaseSchema>HR</DatabaseSchema>
  </LeftDatabase>
  <RightDatabase>
    <ConnectionName>Oracle Schema 2</ConnectionName>
    <ConnectionType>Basic</ConnectionType>
    <Username>HR_TARGET</Username>
    <Password>*****</Password>
    <Host>host-name-or-ip</Host>
    <Service>oracle-service-name</Service>
    <DatabaseSchema>HR_TARGET</DatabaseSchema>
  </RightDatabase>
  <CommandLineOptions>
    <Direction>RightAsLeft</Direction>
```

```

    <Synchronize>true</Synchronize>
    <DataScriptFile>SyncScript.sql</DataScriptFile>
    <DataLogFile>SyncScript.log</DataLogFile>
    <ComparisonLogFile>Operational.txt</ComparisonLogFile>
    <ErrorLogName>Error.log</ErrorLogName>
  </CommandLineOptions>
</DataCompareCmdConfig>

```

- <LeftDatabase> and <RightDatabase> elements specify the source and the target database or schema.
- DatabaseSchema element within each database element specifies the Oracle schema.
- Data compare options are not set. The command line will use the default options.
- The <CommandLineSettings> element contains the following settings:
 - <Direction>: RightAsLeft. Indicates that data on the right will be modified to match with the data on the left.
 - <Synchronize> true: Indicates that the command line should execute the data synchronization script.
 - <DataScriptFile>. Specifies a file name for data synchronization script.
 - <DataLogFile>. Specifies a file name for data synchronization log.
 - <ComparisonLogFile>. Specifies a file name for data compare operational log.
 - <ErrorLogName>. Specifies a file name for logging the errors that might occur during the comparison.

You can run the command line with a xml configuration file as follows:

```
Idera.OracleDataCmd <config.xml>
```

You can validate an xml file as follows:

```
Idera.OracleDataCmd <config.xml> /v
```

Customizing Data Compare

There are various comparison options that you can customize via the command line. You can:

- Change data comparison options.
- Manually map tables or views.
- Exclude tables, views, or columns from the comparison.
- Create custom comparison keys.
- Specify where clauses for tables or views.
- Set mapping rules.

The following sections describe the most common scenarios in details.

- [Data Compare Modifying Comparison Options](#)
- [Data Compare Changing Mapping Type](#)
- [Data Compare Excluding Database Objects](#)
- [Data Compare Excluding Columns](#)
- [Data Compare Defining Custom Keys](#)

- [Data Compare Defining Mapping Rules](#)
- [Data Compare Specifying WHERE Clauses](#)

Data Compare Modifying Comparison Options

Comparison options can be specified under the xml element <DataCompareOptions>.

The following xml fragment specifies two additional comparison option: GenerateDetailedLog

```
<?xml version="1.0" encoding="utf-8" ?>
<DataCompareCmdConfig xmlns="http://www.idera.com/oracle/schemas/datacompare/cmd.xsd">
  <LeftDatabase>
    <ConnectionName>Oracle Schema 1</ConnectionName>
    <ConnectionType>Basic</ConnectionType>
    <Username>HR</Username>
    <Password>*****</Password>
    <Host>host-name-or-ip</Host>
    <Service>oracle-service-name</Service>
    <DatabaseSchema>HR</DatabaseSchema>
  </LeftDatabase>
  <RightDatabase>
    <ConnectionName>Oracle Schema 2</ConnectionName>
    <ConnectionType>Basic</ConnectionType>
    <Username>HR_TARGET</Username>
    <Password>*****</Password>
    <Host>host-name-or-ip</Host>
    <Service>oracle-service-name</Service>
    <DatabaseSchema>HR_TARGET</DatabaseSchema>
  </RightDatabase>
  <DataCompareOptions>
    <GenerateDetailedLog>>true</GenerateDetailedLog>
  </DataCompareOptions>
  <CommandLineOptions>
    <Direction>RightAsLeft</Direction>
    <Synchronize>>true</Synchronize>
    <DataScriptFile>SyncScript.sql</DataScriptFile>
    <DataLogFile>SyncScript.log</DataLogFile>
    <ComparisonLogFile>Operational.txt</ComparisonLogFile>
    <ErrorLogName>Error.log</ErrorLogName>
  </CommandLineOptions>
</DataCompareCmdConfig>
```

For the full list of supported options, check Xml Reference.

Data Compare Changing Mapping Type


Mapping determines how database tables and views are paired together. Default mapping is based on the name of the object. A table/view on the "left" schema can only be mapped to a table/view with the same name on the "right" schema. You can change the mapping type by choosing a "Custom" mapping and specify the objects that should be paired together, which might not necessarily have the same name.

To define custom mappings, you need to:

- Set the <TableMappingSchema> under <CommandLineOptions> element to Custom.
- Specify the custom pairs of Tables/Views under the element <TablePairs> or <ViewPairs>

The following xml fragment defines custom mappings:

```
<CommandLineOptions>
  <TableMappingSchema>Custom</TableMappingSchema>
</CommandLineOptions>
<TablePairs>
  <Pair>
    <LeftObject Name="dbo.Table_A"></LeftObject>
    <RightObject Name="dbo.Table_AA"></RightObject>
  </Pair>
  <Pair>
    <LeftObject Name="dbo.Table_B"></LeftObject>
    <RightObject Name="dbo.Table_BB"></RightObject>
  </Pair>
</TablePairs>
```

 Specify the full name of the objects in the custom mappings set.

Data Compare Excluding Database Objects

You can exclude specific tables or views from data compare by listing them under the xml element <TablePairs> or <ViewPairs>.

Rules for excluding database objects are as follows:

- The attribute Action of <Pair> element should be set to Exclude
- The left and right object must be both specified for each excluded pair

The following xml fragment excludes the tables HR.EMPLOYEES and HR_COPY.EMPLOYEES from the comparison:

```
<TablePairs>
  <Pair Action="Exclude">
    <LeftObject Name="HR.EMPLOYEES"></LeftObject>
    <RightObject Name="HR_COPY.EMPLOYEES"></RightObject>
  </Pair>
  <Pair Action="Exclude">
    <LeftObject Name="HR.EMPLOYEES"></LeftObject>
    <RightObject Name="HR_COPY.EMPLOYEES"></RightObject>
  </Pair>
</TablePairs>
```

```

    </Pair>
</TablePairs>

```

Data Compare Excluding Columns

Command line allows you to exclude specific columns from data compare.

Rules for excluding columns are as follows:

- Excluded columns should be listed under the xml element `<ColumnPair>` of the `<Columns>` element
- The attribute `Action` of the `<Columns>` element should be set to `Exclude`

The following xml fragment excludes the columns `FIRST_NAME`, `LAST_NAME` from the table `HR.EMPLOYEES` and the column `ADDRESS` from the table `HR.CUSTOMERS`:

```

<TablePairs>
  <Pair>
    <LeftObject Name="HR.EMPLOYEES"></LeftObject>
    <RightObject Name="HR_COPY.EMPLOYEES"></RightObject>
    <Columns Action="Exclude">
      <ColumnPair>
        <LeftColumn>FIRST_NAME</LeftColumn>
        <RightColumn>FIRST_NAME</RightColumn>
      </ColumnPair>
      <ColumnPair>
        <LeftColumn>LAST_NAME</LeftColumn>
        <RightColumn>LAST_NAME</RightColumn>
      </ColumnPair>
    </Columns>
  </Pair>
  <Pair>
    <LeftObject Name="HR.CUSTOMERS"></LeftObject>
    <RightObject Name="HR_COPY.CUSTOMERS"></RightObject>
    <Columns Action="Exclude">
      <ColumnPair>
        <LeftColumn>ADDRESS</LeftColumn>
        <RightColumn>ADDRESS</RightColumn>
      </ColumnPair>
    </Columns>
  </Pair>
</TablePairs>

```

Data Compare Defining Custom Keys

You can specify a custom key when the table or view has no built-in key or when the existing key is not suitable for data compare.

The following rules apply to custom keys:

- A custom key can be defined under the xml element <LeftObject> or <RightObject> of the <Pair> element
- A custom key must have a name and one or more columns
- The columns that participate in the custom key should exist in the parent table or view
- The key must be defined for each object in the pair of entities.
- The columns that participate in the custom key must be identical for both objects in the specified pair.

The following xml fragment defines a custom key for the pair of table HR.EMPLOYEES:

```
<TablePairs>
  <Pair>
    <LeftObject Name="HR.EMPLOYEES">
      <Key Name="Custom_Key">
        <Column>EMPLOYEE_ID</Column>
      </Key>
    </LeftObject>
    <RightObject Name="HR_COPY.EMPLOYEES">
      <Key Name="Custom_Key">
        <Column>EMPLOYEE_ID</Column>
      </Key>
    </RightObject>
  </Pair>
</TablePairs>
```

Data Compare Defining Mapping Rules

Mapping rules are used to override the default data compare mapping that is based on name.

The following xml fragment defines a mapping rule that ignores the name postfix "_temp" on the left schema:

```
<TablesMappingRules>
  <NameMapping MappingRule="IgnorePostfix" IgnoreLeftName="_temp" />
</TablesMappingRules>
```

Data Compare Specifying WHERE Clauses

A where clause is an expression that is used as a data filter. Only the records that meet the where clause criteria are compared and synchronized. You can specify any valid expression as the where clause. The "where" keyword should not be included in the expression.

The command line supports where clauses via three elements under the <Pair> element:

- WhereClause: sets a where clause for both members of a pair
- LeftWhereClause: sets a where clause for the left member of the pair
- RightWhereClause: sets a where clause for the right member of the pair

The following xml fragment defines a where clauses on the employees table that excludes records with emp_id = 3. The where clause is set for both the left and the right schema:

```
<TablePairs>
  <Pair>
    <LeftObject Name="HR.EMPLOYEES"/>
    <RightObject Name="HR_COPY.EMPLOYEES"/>
    <WhereClause>EMPLOYEE_ID != 3</WhereClause>
  </Pair>
</TablePairs>
```

The following xml fragment defines separate where clauses on the employees table for the left and the right schema. On the left schema, it excludes records with EMPLOYEE_ID = 3; on the right schema, it excludes records with EMPLOYEE_ID = 4.

```
<TablePairs>
  <Pair>
    <LeftObject Name="HR.EMPLOYEES"/>
    <RightObject Name="HR_COPY.EMPLOYEES"/>
    <LeftWhereClause>EMPLOYEE_ID != 3</LeftWhereClause>
    <RightWhereClause>EMPLOYEE_ID != 4</RightWhereClause>
  </Pair>
</TablePairs>
```

Data Compare XML Configuration File Reference

```
<DataCompareCmdConfig>
  <LeftDatabase>
    <ConnectionName>string</ConnectionName>
    <OracleAuthentication>boolean</OracleAuthentication>
    <Username>string</Username>
    <Password>string</Password>
    <DatabaseSchema>string</DatabaseSchema>
    <ConnectionType>ConnectionTypes</ConnectionType>
    <Host>string</Host>
    <Port>int</Port>
    <Service>string</Service>
    <DataSource>string</DataSource>
    <DbPrivilege>DbPrivileges</DbPrivilege>
    <ConnectionTimeout>int</ConnectionTimeout>
    <CommandTimeout>int</CommandTimeout>
  </LeftDatabase>
  <RightDatabase>
    <ConnectionName>string</ConnectionName>
    <OracleAuthentication>boolean</OracleAuthentication>
    <Username>string</Username>
    <Password>string</Password>
    <DatabaseSchema>string</DatabaseSchema>
```

```

<ConnectionType>ConnectionTypes</ConnectionType>
<Host>string</Host>
<Port>int</Port>
<Service>string</Service>
<DataSource>string</DataSource>
<DbPrivilege>DbPrivileges</DbPrivilege>
<ConnectionTimeout>int</ConnectionTimeout>
<CommandTimeout>int</CommandTimeout>
</RightDatabase>

<DataCompareOptions>
  <CompareTables>boolean</CompareTables>
  <CompareViews>boolean</CompareViews>
  <DisableTriggers>boolean</DisableTriggers>
  <ForeignKeyAction>ForeignKeyActions</ForeignKeyAction>
  <GenerateDetailedLog>boolean</GenerateDetailedLog>
  <LogScriptExecution>boolean</LogScriptExecution>
  <ScriptExecutionLogFile>string</ScriptExecutionLogFile>
  <GenerateTransactionalScript>boolean</GenerateTransactionalScript>
  <IgnoreCharCase>boolean</IgnoreCharCase>
  <TrimSpaces>boolean</TrimSpaces>
  <PerformDependencyAnalysis>boolean</PerformDependencyAnalysis>
  <ReportEqualRows>boolean</ReportEqualRows>
  <SynchronizeDifferentRows>boolean</SynchronizeDifferentRows>
  <SynchronizeLeftRows>boolean</SynchronizeLeftRows>
  <SynchronizeRightRows>boolean</SynchronizeRightRows>
  <TransactionSize>int</TransactionSize>
  <RoundApproximateNumber>boolean</RoundApproximateNumber>
  <RoundPrecision>int</RoundPrecision>
</DataCompareOptions>

<CommandLineOptions>
  <ComparisonLogFile>string</ComparisonLogFile>
  <OverwriteComparisonLog>boolean</OverwriteComparisonLog>
  <ErrorLogName>string</ErrorLogName>
  <ErrorLogType>ErrorLogTypes</ErrorLogType>
  <QuietMode>boolean</QuietMode>
  <DataScriptFile>string</DataScriptFile>
  <DataLogFile>string</DataLogFile>
  <FileEncoding>Encodings</FileEncoding>
  <Direction>CompareDirections</Direction>
  <Synchronize>boolean</Synchronize>
  <TableMappingSchema>TableMappingSchemaTypes</TableMappingSchema>
</CommandLineOptions>

<TablesMappingRules>
  <NameMapping>NameMappingType</NameMapping>
</TablesMappingRules>

```

```

<ViewsMappingRules>
  <NameMapping>NameMappingType</NameMapping>
</ViewsMappingRules>

<TablePairs>Pairs</TablePairs>
<ViewPairs>Pairs</ViewPairs>

<NameMappingType>
  <MappingRule>NameMappingRules</MappingRule>
  <IgnoreLeftName>string</IgnoreLeftName>
  <IgnoreRightName>string</IgnoreRightName>
</NameMappingType>

<NameMappingRules>ExactMatch|IgnorePrefix|IgnorePostfix</NameMappingRules>
<SchemaPairs>[1..*]SchemaPairType</SchemaPairs>
<SchemaPairType>
  <LeftSchema>string</LeftSchema>
  <RightSchema>string</RightSchema>
</SchemaPairType>

<Pairs>[0..*]Pair</Pairs>
<Pair>
  <LeftObject>TableOrView</LeftObject>
  <RightObject>TableOrView</RightObject>
  <Columns>Columns</Columns>
  <WhereClause>string</WhereClause>
  <Action>Actions</Action>
</Pair>
<TableOrView>
  <Key>DataKey</Key>
  <Name>string</Name>
</TableOrView>
<DataKey>
  <Name>string</Name>
  <Column>[1..*]string</Column>
</DataKey>
<Columns>
  <Action>Actions</Action>
  <ColumnPair>[1..*]ColumnPair</ColumnPair>
</Columns>
<Actions>Include|Exclude</Actions>
  <ColumnPair>
    <LeftColumn>string</LeftColumn>
    <RightColumn>string</RightColumn>
  </ColumnPair>

<ConnectionTypes>Basic|TnsIdentifier|EasyConnect|ConnectDescriptor</ConnectionTypes>
<DbPrivileges>Default|SysDba|SysOper</DbPrivileges>

```

```

<ForeignKeyActions>DisableAndEnable|DropAndCreate|None</ForeignKeyActions>
<ErrorLogTypes>File|WindowsLog</ErrorLogTypes>
<Encodings>Unicode|UTF7|UTF8|ASCII|ANSI</Encodings>
<CompareDirections>LeftAsRight|RightAsLeft</CompareDirections>
<TableMappingSchemaTypes>Auto|Custom</TableMappingSchemaTypes>

```

```
</DataCompareCmdConfig>
```

The main elements of xml schema are described in the following section:

Element	Description
DataCompareCmdConfig	Root element.
LeftDatabase, RightDatabase	The elements that specify the left and the right database that participate in the data comparison.
ConnectionName	Specifies the name of the SQL Server instance.
OracleAuthentication	If true, the connection uses Oracle authentication.
Username, Password	Define database credentials if the authentication is set to OracleAuthentication.
DatabaseSchema	Defines the schema that should be compared. If not specified, the schema associated with the connection will be used.
ConnectionType	Indicates the Oracle connection type: Basic, TnsIdentifier, EasyConnect or ConnectDescriptor.
Host	Defines the Oracle host name or IP. Host is considered when the connection type is "Basic".
Port	Defines the port number. If not specified, the default port 1521 will be used. Port is considered only when the connection type is "Basic".
Service	Defines the Oracle service name. Service name is considered only when the connection type is "Basic".
DataSource	Defines the connection string for the Oracle database. DataSource is considered only when the connection type is TnsIdentifier, EasyConnect or ConnectDescriptor.
DbPrivilege	Defines the privileges associated with the connection.

Element	Description
ConnectionTimeout	Defines the connection timeout in seconds, which the amount of time the command line waits before aborting a connection.
CommandTimeout	Defines the execution timeout of Oracle queries in seconds, which the amount of time the command line waits for a query to complete before aborting it.
CompareTables	Indicates whether the Oracle tables should be compared and synchronized.
CompareViews	Indicates whether Oracle views should be compared and synchronized.
DisableTriggers	Indicates whether the triggers should be disabled before executing the synchronization script.
ForeignKeyAction	Specifies the type of action on foreign keys that should be performed by synchronization script.
GenerateDetailedLog	Indicates whether the command line should generate a detailed synchronization log.
LogScriptExecution	Specifies whether the command line should save to a log file statements executed during the synchronization.
ScriptExecutionLogFile	Defines the file name of the execution log.
GenerateTransactionalScript	Indicates whether the synchronization script should be transactional.
IgnoreCharCase	Indicates whether the character chase should be ignored when comparing string columns.
TrimSpaces	Indicates whether the leading and trailing spaces should be ignored when comparing string columns.
PerformDependencyAnalysis	Specifies whether command line should analyze object dependencies (based on foreign key relationship) and order the synchronization script accordingly.
ReportEqualRows	Indicates whether the command line should keep the equal rows in the internal row collections. By default equal rows are discarded.
SynchronizeDifferentRows	Indicates whether the rows that have been found different should be synchronized.

Element	Description
SynchronizeLeftRows	Indicates whether the rows found on the left database should be synchronized.
SynchronizeRightRows	Indicates whether the rows found on the right database should be synchronized.
TransactionSize	Defines the max size of the Oracle transaction.
RoundApproximateNumber	Indicates whether approximate should be rounding to a specified precision and then compared.
RoundPrecision	Defines the rounding precision for approximate data types such as real and float.
ComparisonLogFile	Defines a file name for data compare operational messages.
OverwriteComparisonLog	Indicates whether the command line should overwrite all log files. By default command line appends to log file.
ErrorLogName	Defines a file name for the error log.
ErrorLogType	Indicates whether errors should be logged to file or Windows event log.
QuietMode	Indicates the quiet mode which suppresses messages from being displayed on the console.
DataScriptFile	Defines a file name for the data synchronization script.
DataLogFile	Defines a file name for the data synchronization log.
FileEncoding	Defines the encoding of all output file.
Direction	Indicates the "direction" of synchronization whether the left schema is synchronized with the right schema or vice versa.
Synchronize	Indicates whether the command line should synchronize the data. By default, data is compared, but not synchronized.
TableMappingSchema	Defines the mapping type of schema objects: Auto or Custom.
TablesMappingRules	Defines the mapping rules for tables.
ViewsMappingRules	Defines the mapping rules for views.

Element	Description
NameMapping	Defines the mapping rule that applies to objects name.
TablePairs	Defines that pair of tables that should be mapped manually. Manual pairs require the option TableMappingSchema to be Custom.
ViewPairs	Defines that pair of views that should be mapped manually. Manual pairs require the option TableMappingSchema to be Custom.

Data Compare Error Codes

Data Compare command line returns the following error codes:

Error Code	Description
0	No error
1	Invalid command line switches
2	Invalid xml file
4	Invalid output files
8	Invalid or missing options
16	Generic error
32	Missing or expired license

Comparison Toolset for Oracle Database PDF

This page contains a direct link to the IDERA Comparison Toolset for Oracle Database help in PDF format. This format is suitable for printing and saving on your local machine for further reference. The PDF includes all pages from the relevant product help published on wiki.idera.com.

- [IDERA Comparison Toolset for Oracle Database 8.0 PDF](#)