

DB Change Manager 18.0

Quick Start Guide

Exported on 09/17/2021

I D E R A

Table of Contents

- Welcome to DB Change Manager5**
 - What can you do with DB Change Manager?5
- Getting started7**
 - Data comparison and synchronization7
 - Schema comparison and synchronization7
 - Database configuration management7
 - Reports, notifications, and change history7
 - Data masking7
 - Source control system integration8
 - Product requirements and database support8
 - Eclipse plugin support9
- Installing DB Change Manager9**
 - Installing DB Change Manager in an AppWave environment9
- Licensing DB Change Manager10**
 - Registering a Trial or Workstation license at application startup10
 - Registering a Workstation license after application startup10
 - Registering by phone11
- Key concepts of DB Change Manager11**
 - Understanding jobs and scripts in DB Change Manager11
 - Understanding the DB Change Manager User Interface12
- Tutorials for DB Change Manager13**
 - Tutorial 1: Comparing data13
 - Adding data sources14
 - Creating a data comparison job16
 - Tutorial 2: Synchronizing data21
 - Highlighted Feature: The Compression Comparison Option22
 - Tutorial 3: Comparing configurations23
 - Highlighted feature: Comparing multiple targets24
 - Tutorial 4: Defining a configuration standard24
 - Tutorial 5: Comparing and synchronizing schemas25
 - Tutorial 6: Defining a schema archive26

Tutorial 7: Masking data 27

Tutorial 8: Defining and running automated scripts 29

Use DB Change Manager to:

- Quickly roll out and reconcile database changes
- Reveal, track, and report on database changes
- Comply with database audit and reporting requirements
- Protect data privacy within the database environment
- Track changes from multiple major database platforms

Welcome to DB Change Manager

IDERA DB Change Manager helps database administrators and developers to:

- Simplify and automate database change management.
- Streamline development cycles.
- Ensure availability, performance, and compliance.

DB Change Manager provides comparison, synchronization, data masking, and auditing capabilities that let you track and report on changes, roll out new releases, and pinpoint problems resulting from changes at the data, schema, and database configuration levels.

By comparing a live database to a schema or configuration “snapshot,” administrators can quickly identify changes and correct problems. By monitoring configuration settings, administrators can ensure compliance with regulatory policies and performance standards and maintain overall database performance and availability.

What can you do with DB Change Manager?

This topic describes what you can do with DB Change Manager and links to more detailed information about how to perform the tasks. In addition, DB Change Manager includes step-by-step Eclipse-style cheat sheets to help you learn the tasks.

- **Complete Commonly-performed tasks.** These tasks describe basic functions and help you use DB Change Manager most effectively. In addition, you can customize the interface using preferences. See the [Preference reference](#) for details.
 - [Registering and organizing data sources](#)
 - [Creating a job report](#)
 - [Scheduling a job](#)
 - [Filtering data source objects](#)
 - [Searching for database objects and synchronization scripts](#)
 - [Viewing Change History](#)
 - [Exporting and importing jobs](#)
- **Compare and synchronize data.** Data Comparison jobs help you manage data changes and synchronize changes between databases.
 - [Comparing data](#)
 - [Synchronizing data repositories](#)
- **Compare and synchronize schemas.** In DB Change Manager you can select schema objects and types, compare them across data sources, and synchronize them to each other. You can also create a snapshot or archive of the schema objects, to use in a comparison or to save for future reference. In addition, you can clone schema comparison jobs to make it easier to reuse existing jobs.
 - [Comparing schemas](#), including [how to clone a schema comparison job](#)
 - [Synchronizing schemas](#)
 - [Creating a schema archive](#)
- **Manage database configurations.** DB Change Manager helps you keep track of database settings and version information; such as patch levels. You can define database configuration values and ranges and then run a configuration comparison job to check for compliance against those parameters.
 - [Comparing database configuration settings](#)
 - [Creating a standard for configuration settings](#)
 - [Creating a configuration archive](#)
- **Create masked data for development environments.** Protecting personally identifiable information while having accurate data for testing can be difficult. DB Change Manager lets you create a masked copy of production data for use during development and testing. This way the data is functionally correct and at the same time does not compromise privacy or security.

- [Masking data](#)
- **Automate and script jobs.** You may need some jobs to run on a regular basis without using Windows Task Scheduler (as described in [Scheduling a job.](#)) The Script Wizard lets you define scripts that run jobs, and batch files to run multiple scripts.
 - [Scheduling a job](#)
 - [Generating command line scripts](#)

Getting started

DB Change Manager provides features to help you manage databases from development to deployment and maintenance cycles. These features include:

- [Data comparison and synchronization](#)
- [Schema comparison and synchronization](#)
- [Database configuration management](#)
- [Reports, notifications, and change history](#)
- [Data masking](#)
- [Source control system integration](#)

Data comparison and synchronization

DB Change Manager can analyze and compare the data in two different data sources. It generates a SQL script that can update one data source so that it matches another. You can modify the SQL and run it as you wish. You can even define a comparison and synchronization process, called a job, and set it to run automatically. These features help you better manage common scenarios, such as:

- Keeping primary and backup data in sync
- Keeping up with data changes during the development and testing cycles, and later on to deployment environments
- Managing data migrations during upgrade activities

Schema comparison and synchronization

DB Change Manager can capture and compare the database schema in your enterprise. Over the lifecycle of a single schema, you can capture a baseline or archive schema and analyze changes through time. In addition, DB Change Manager generates a SQL script that can update a target schema so that it matches a standard. You can modify the SQL and execute it as needed.

Database configuration management

In addition to schema, there are configuration settings that can vary across an enterprise. DB Change Manager can capture the configuration settings of a data source. You can capture a baseline or archive configuration and analyze changes through time. You can create a configuration standard that defines the norms for your enterprise and against which you can compare your databases.

Reports, notifications, and change history

DB Change Manager supports your data management activities with notifications when processes execute, reports that capture key information, and change history tracking.

Data masking

There are specific scenarios when you need to hide the true values of the data in a database. For example, during development you need to protect personal or proprietary data, yet you still need to work with data that accurately

represents the source. By combining data synchronization with data masking features, such as randomization or shuffling, you can duplicate source data and use the masked copies in non-production or non-secure environments.

Source control system integration

You can integrate DB Change Manager with source control systems such as Concurrent Versions System (CVS), Visual Source Safe (VSS), or Eclipse plugins. The Navigator in DB Change Manager is the interface for accessing files under source control. Job files and archive versions are stored as XML files and synchronization scripts are stored as text files. You perform all source code control actions using the source control system.

Product requirements and database support

Type	Platform	Version
CPU		<ul style="list-style-type: none"> • 1+ GHz
Random Access Memory (RAM)		<ul style="list-style-type: none"> • 3 GB
Storage Disk Space		<ul style="list-style-type: none"> • 1 GB
Operating System	Microsoft Windows	<ul style="list-style-type: none"> • 10 • 8.1 • 8 • 7
	Microsoft Windows Server	<ul style="list-style-type: none"> • 2019 • 2016 • 2012 R2 • 2012 • 2008 SP1 • 2008 R2
Cloud	<ul style="list-style-type: none"> • Amazon Elastic Compute Cloud (EC2) • Microsoft Azure Virtual Machine (VM) 	
Database	IBM Db2 LUW and Z/OS	<ul style="list-style-type: none"> • 11.x* • 10.x

Type	Platform	Version
	Microsoft SQL Server	<ul style="list-style-type: none"> • Azure* • 2019* • 2017* • 2016* • 2014 • 2012 SP3 • 2008
	Oracle	<ul style="list-style-type: none"> • 21c* • 19c* • 18c* • 12c R2* • 11g R2* • 10g*
	Sybase ASE	<ul style="list-style-type: none"> • 16.x* • 15.x

* DB Change Manager supports only a subset of features and functions for this database version.

 DB Change Manager supports data sources that use the Unicode format.

Eclipse plugin support

If you choose to install the Eclipse plugin version, you also need the following items:

- Eclipse 3.6+
- Sun Java Standard Edition 6, Update 45+

Installing DB Change Manager

To install DB Change Manager on a workstation, run the installer and follow the prompts provided by the Installation Wizard. When the installation is complete, see the ReadMe file for known issues or advisory notes that could affect your use of the product.

Installing DB Change Manager in an AppWave environment

If you are a DB Change Manager user and your System Administrator provides you with AppWave, then your System Administrator enables DB Change Manager in your AppWave Browser. Consult your System Administrator and the AppWave Browser online help for details.

Licensing DB Change Manager

DB Change Manager supports the following types of licenses:

- **Trial licenses.** A license for a time-limited, full-featured trial version of the product. You must register the trial license before you can use the product.
- **Workstation licenses.** A license or set of licenses that is tied to a particular workstation. The product can only be used on that workstation.
- **AppWave licenses.** AppWave licenses are normally concurrent licenses managed by your System Administrator. See your System Administrator for details.

Registering a Trial or Workstation license at application startup

Shortly after downloading a trial version of an IDERA product, you should receive an email from IDERA containing a serial number, which must be registered. These instructions also apply if you have purchased a product and received a serial number prior to running the application for the first time, or if the trial period expired.

To register a serial number at application startup

1. Start the application. The License Registration dialog appears.
2. Copy the serial number from the email, and then paste it into **Serial Number**.
3. Enter your Developer Network (EDN) account credentials in the **Login**, or in the **Email** and **Password** fields. If you have not previously created an EDN account, or have forgotten your password, click **I need to create ... or I've lost my password**.
4. Click **Register**.

The activation file should automatically download and install. If this does not happen, click the **Trouble Connecting? Try Web Registration** link and follow the prompts.

Registering a Workstation license after application startup

The following instructions assume that you have received a workstation license by email and that you currently have a valid trial license. If you did not install a trial version or if the trial period has expired, follow the instructions in [Registering a Trial or Workstation License at Application Startup](#) instead.

To register a serial number after application startup

1. Click **Help > Launch License Manager**.
The License Manager dialog appears.
2. In the License Manager dialog, click **Serial > Add**.
3. Copy the serial number from the email, and then paste it into the Add Serial Number dialog.
4. Click **OK**.
5. Expand Unregistered Serial Numbers, right-click the serial number just added, and then click **Register**.
A registration dialog appears.

 The Registration Code box shows a machine-specific identifier required with other registration methods.

6. Ensure that **Register using Online Registration** is selected.
7. Provide Developer Network credentials in the **Login** field, or into the **Email** and **Password** fields.
8. Click **Register**.
The activation file should download and install automatically. If this does not happen, click the **Trouble Connecting? Try Web Registration** link and follow the prompts.

Registering by phone

If you have problems with either of the above procedures, you can register licenses by phone. You must provide the following information:

- Developer Network credentials;
- registration code displayed in the License Registration dialog that appears when you start an unlicensed application;
- product base license serial number; and
- license serial numbers for any additional features purchased

For North America, Latin America, and Asia Pacific: call (415) 834-3131 option 2, and then follow the prompts. The hours are Monday to Friday, 6:00 A.M. to 6:00 P.M. Pacific time.

For Europe, Africa, and the Middle East: Call +44 (0)1628-684 494. The hours are Monday to Friday, 9 A.M. to 5:30 P.M. U.K. time.

After phoning in, you will receive an email containing an activation file to import into DB Change Manager.

To import an activation file

1. Save the activation file from the email to the desktop or to a scratch directory such as c:\temp.
2. From within the application, click **Help > IDERA Licensing > License Registration**.
3. Click **License > Import**.
4. Navigate to the directory where you saved the activation file and then double-click the activation
5. Click the **Import** button to import the activation
6. Click **Finish**.

Key concepts of DB Change Manager

- [Understanding jobs and scripts in DB Change Manager](#)
- [Understanding the DB Change Manager User Interface](#)

Understanding jobs and scripts in DB Change Manager

DB Change Manager uses the concept of a job to organize reusable processes. A job uses objects of the following types:

- **Data source.** A data source is a data repository or database.
- **Configuration standard.** A standard is a customized set of properties with threshold operators that help you create optimal configurations against which you can compare other data source configurations.
- **Schema or configuration archive.** An archive is a snapshot of a configuration or schema.

You can use this archive to compare against other schemas or configurations.

There are two types of jobs in DB Change Manager:

- **Object creation jobs.** Some jobs create objects that are used in other jobs. You use a job to create configuration standards and archives.
- **Comparison jobs.** Processes source and target objects and provides information describing their similarities and differences. This comparison is then used to generate SQL scripts. A comparison job can be of the following types:

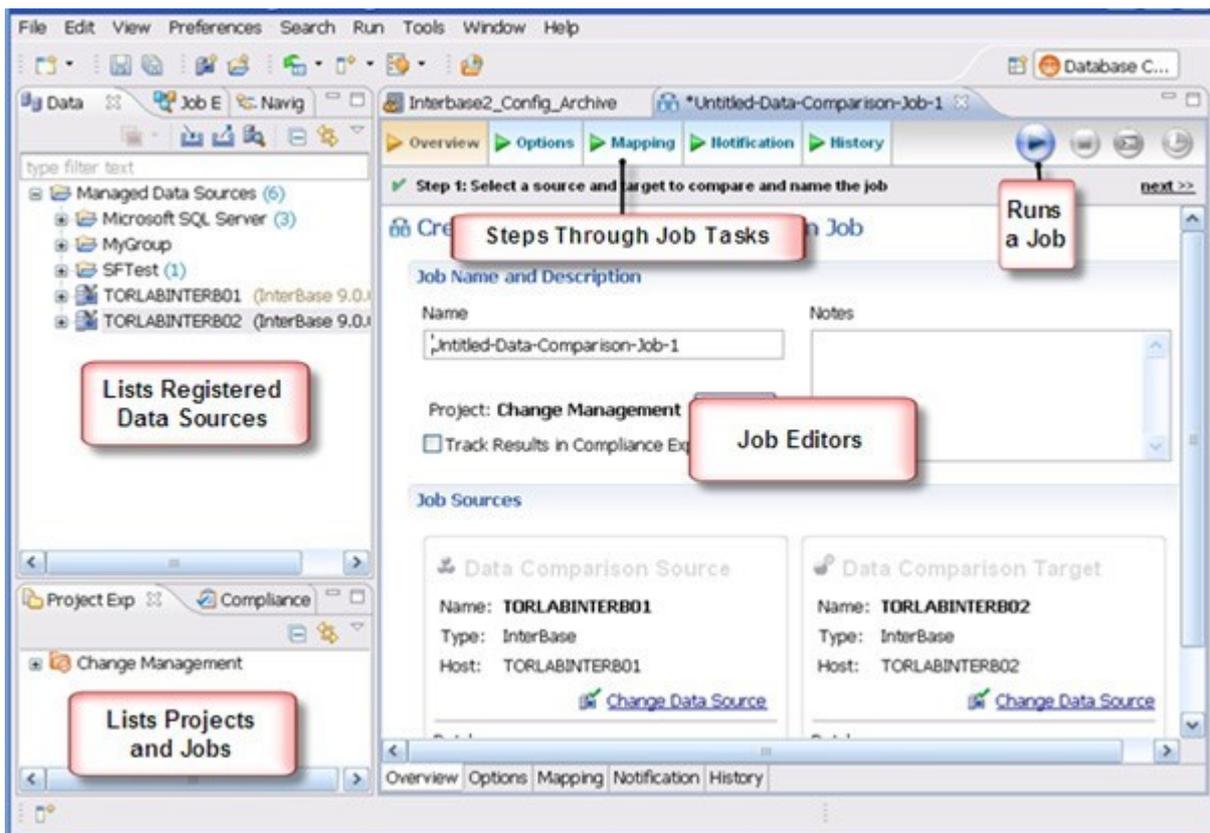
- **Data comparison.** Looks at the data repositories of two data sources and determines their differences. You can use this information to determine whether to update a target data source.
- **Data masking.** A special kind of data comparison job that creates a copy of sensitive data, which masks the true values of specified fields.
- **Configuration comparison.** Determines whether the configuration settings between a source and a target are the same. This information helps you quickly diagnose problems.
- **Schema comparison.** Lets you compare the schemas of different data sources or archives. It helps you manage schema changes through development, testing, and production cycles.

Once it has run a comparison job, DB Change Manager can generate an SQL synchronization script that can update target data, configurations, or schemas. You can modify a script before you run it. For example, you can use a DB Change Manager script to synchronize data between development and testing environments. Once you are comfortable with the job’s specifications, you can let it automatically run the script and synchronize the data.

You can group jobs and scripts into projects, which are visible in the Project explorer and in the Navigator view. The Navigator view provides a way to interact with files in a source control system.

Understanding the DB Change Manager User Interface

The Workbench is the DB Change Manager development environment. It provides you with an interface to create, manage, and navigate comparison jobs through interaction with views, editors, and menus. If you are familiar with Eclipse you will recognize many elements of the interface.



The Welcome Page is the first screen you see when you initially launch the application. The screen provides links to information that may be of value to new users.

Views help you navigate a hierarchy of information, open editors, or display the properties of various application elements. For example, the Data Source Explorer view provides a tree of all data sources in the environment and the comparison jobs associated with each. You can launch these jobs directly, modify the connection properties of data sources, or create and edit configuration archives from the view.

Editors help you perform various tasks, such as running a job. For example, you use the Schema Comparison Job editor to define and execute schema comparison jobs.

The Menu Bar and Command Toolbar options provide various functions, such as launching views and editors, navigating, and setting preferences.

Tutorials for DB Change Manager

 Before working with tutorials, identify two data sources that you can use for practice. These data sources **should not be live or mission-critical**, and they should be similar in schema and data.

This section guides you through the key features of DB Change Manager using the following step-by-step tutorials:

- [Tutorial 1: Comparing data](#)
- [Tutorial 2: Synchronizing data](#)
- [Tutorial 3: Comparing configurations](#)
- [Tutorial 4: Defining a configuration standard](#)
- [Tutorial 5: Comparing and synchronizing schemas](#)
- [Tutorial 6: Defining a schema archive](#)
- [Tutorial 7: Masking data](#)
- [Tutorial 8: Defining and running automated scripts](#)

Tutorial 1: Comparing data

 Before working with tutorials, identify two data sources that you can use for practice. These data sources **should not be live or mission-critical**, and they should be similar in schema and data.

Data comparison jobs automate the process of comparing and correcting data between data sources at the database, table, and row level. You select a source and a target data source from Data Source Explorer, map databases and tables between the two selections, and then run the comparison job.

DB Change Manager compares the information and returns a results screen with details including:

- An overall percentage of the data that is the same between the source and target
- Discrepancies between the two databases, such as data values that exist on only one

Once a data comparison job has run, you can then correct any synchronization issues by running an automatically-generated SQL script that adds missing rows and updates out-of-sync values. You can modify the SQL script prior to running it, and store it for future use.

To start Change Manager

- Start Change Manager using the Start menu's **IDERA DB Change Manager [version]** item.

When DB Change Manager first starts, it displays a Welcome screen. Review the Welcome screen options, and then close it.

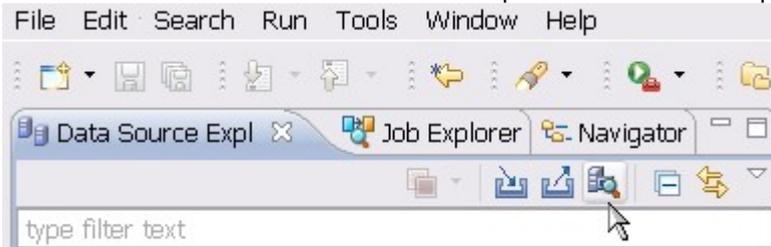
Adding data sources

In order to define and run jobs, you first register data sources by providing connection information and other details to DB Change Manager. Once a data source has been registered, it remains stored in a catalog and does not need to be reregistered each time you open DB Change Manager. You can use a single registered data source in multiple jobs.

Data Source Explorer shows the registered data sources and their associated jobs, if any. Before registering a data source, gather the information about the repository location and connectivity permissions.

To add a data source

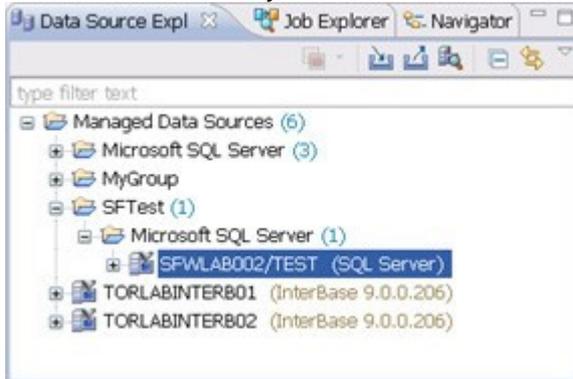
1. Click the Discover Data Sources icon at the top of the Data Source Explorer view.



2. In the Discover Data Sources wizard, select the locations to search for data sources, and then click **Next**.



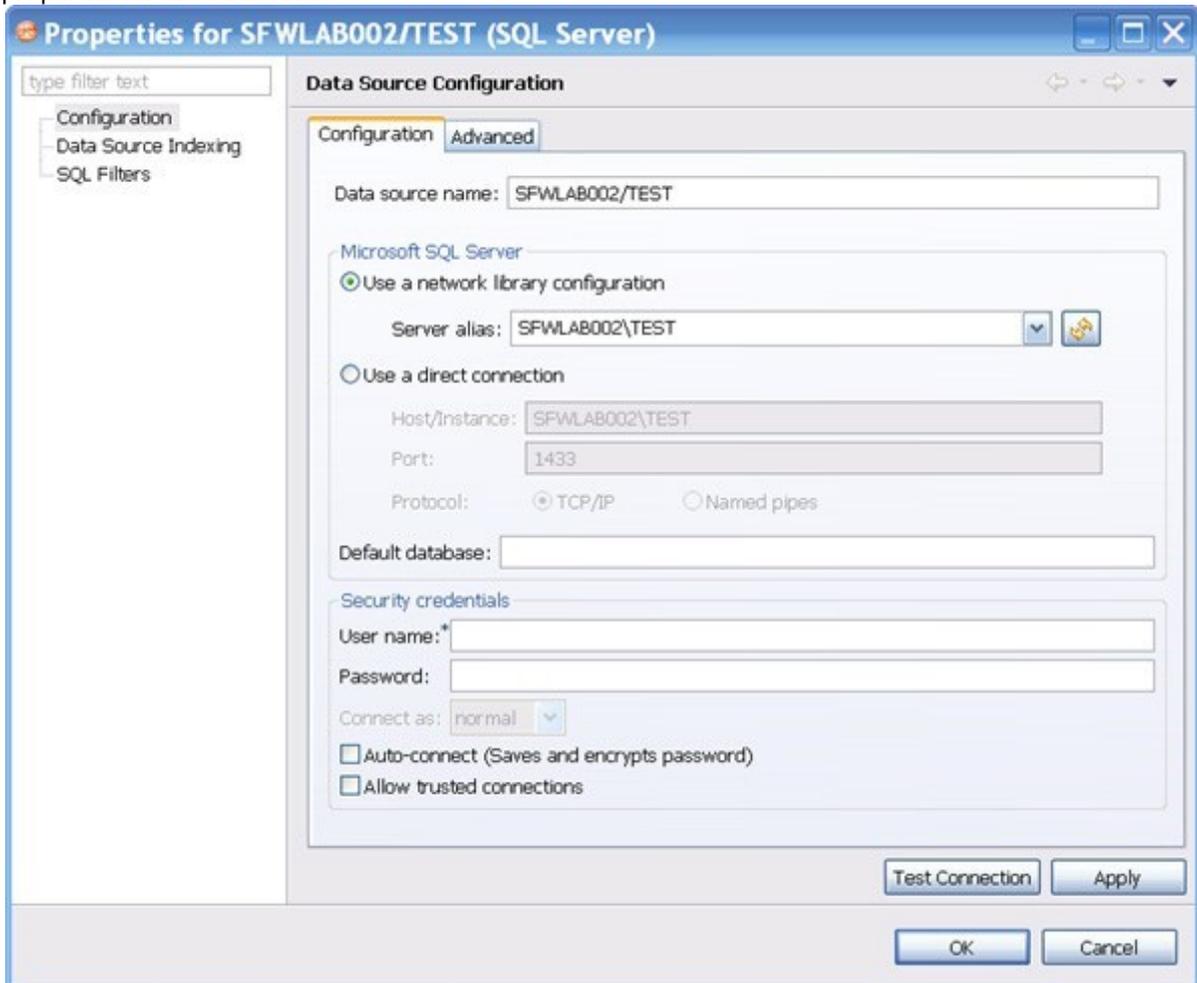
3. Select the data sources you want to add to the DB Change Manager environment, and then click **Finish**.



The new data source or sources appear in the Data Source Explorer.

i An alternative to using the Discover Data Sources wizard is to click **File > New Data Source** and set its connection properties.

4. In the Data Source Explorer, right-click on a data source and select Properties to work with connection properties for the datasource.

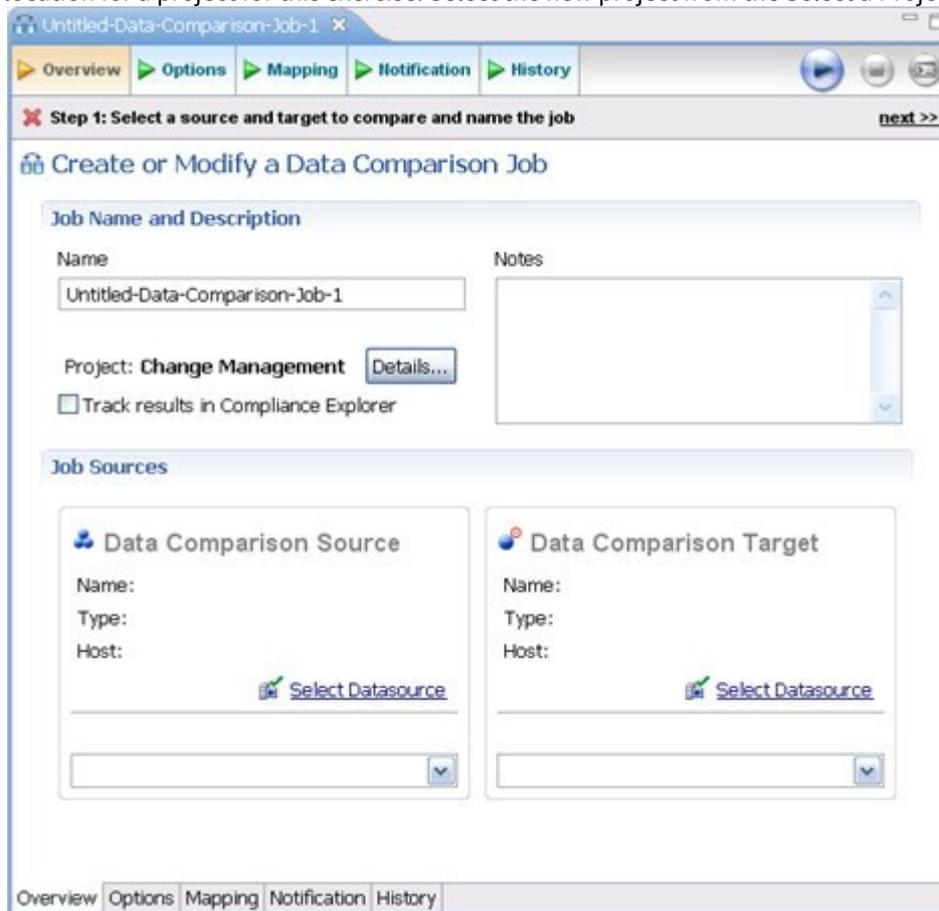


Creating a data comparison job

Prior to performing the following steps, make sure you have at least two data sources registered in DB Change Manager. Ideally these two data sources are similar in schema and content, so that the comparison is meaningful to you.

To create a data comparison job in the Overview tab

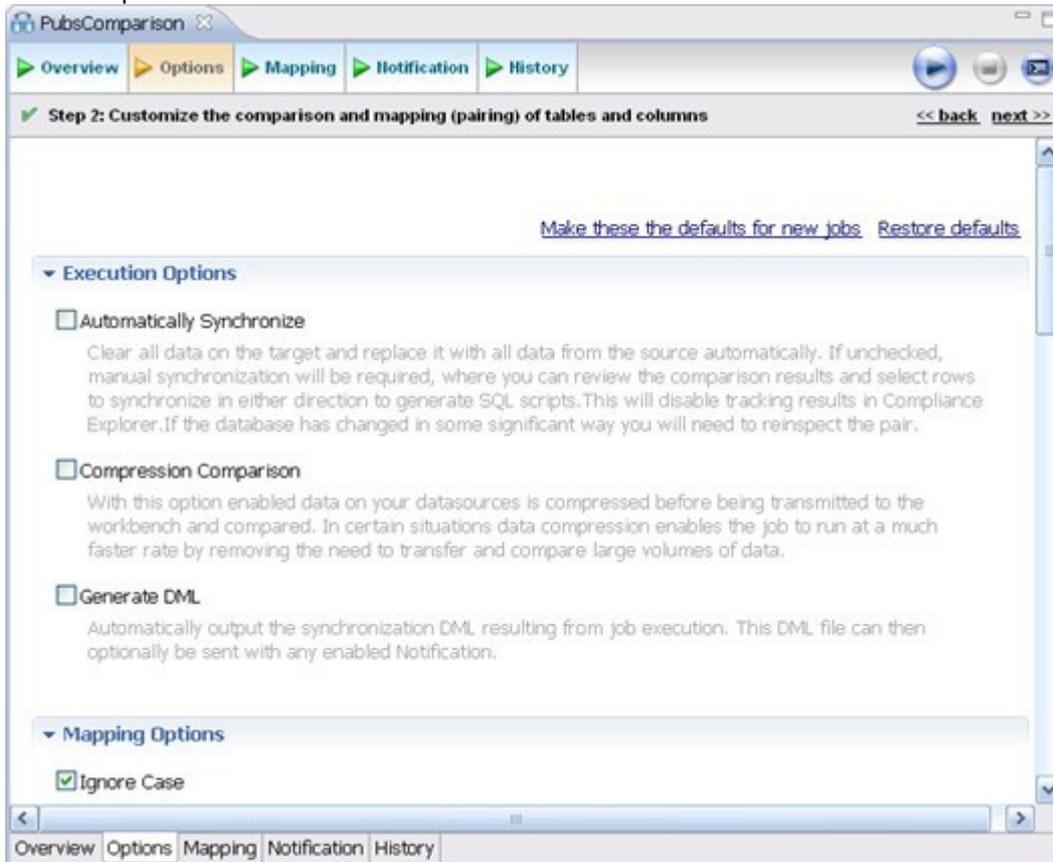
1. Click **File > New > Data Comparison Job**.
2. Use the Select a Project dialog's **Create New** button to open a dialog that lets you provide a name and location for a project for this exercise. Select the new project from the Select a Project dialog.



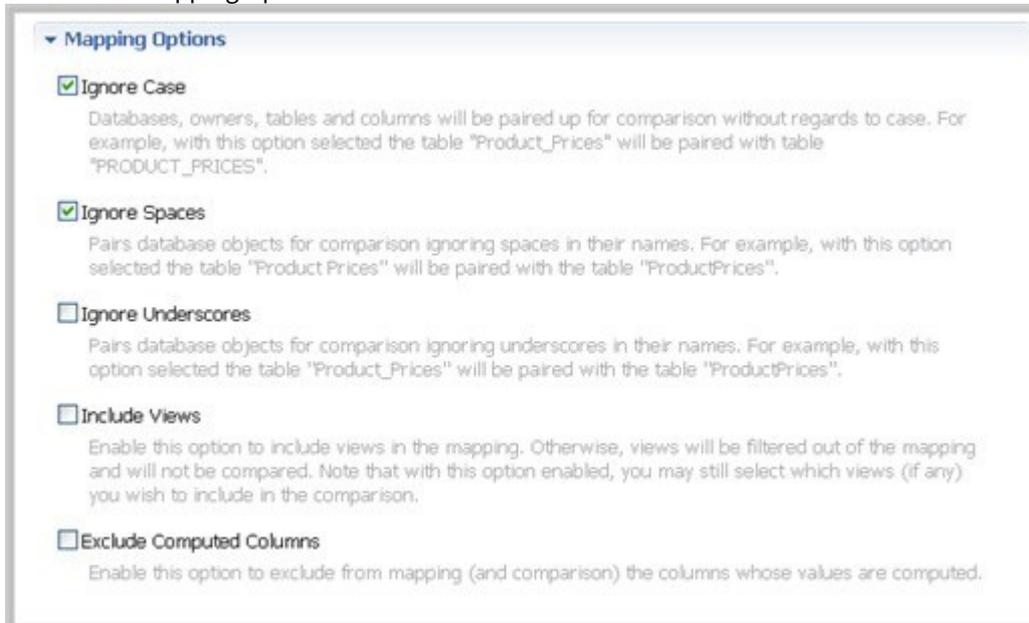
3. In the Overview tab, type the name of the job in the **Name** field.
4. Click and drag a data source from Data Source Explorer to the Data Comparison Source box. By default, all databases in the data source are selected. If you prefer, use the menu to specify a single database.
5. Click and drag a data source from **Data Source Explorer** to the **Data Comparison Target**.
6. Click **File > Save** or type **Ctrl+S** to save the job.

To customize the job in the Options tab

1. Click the Options tab.



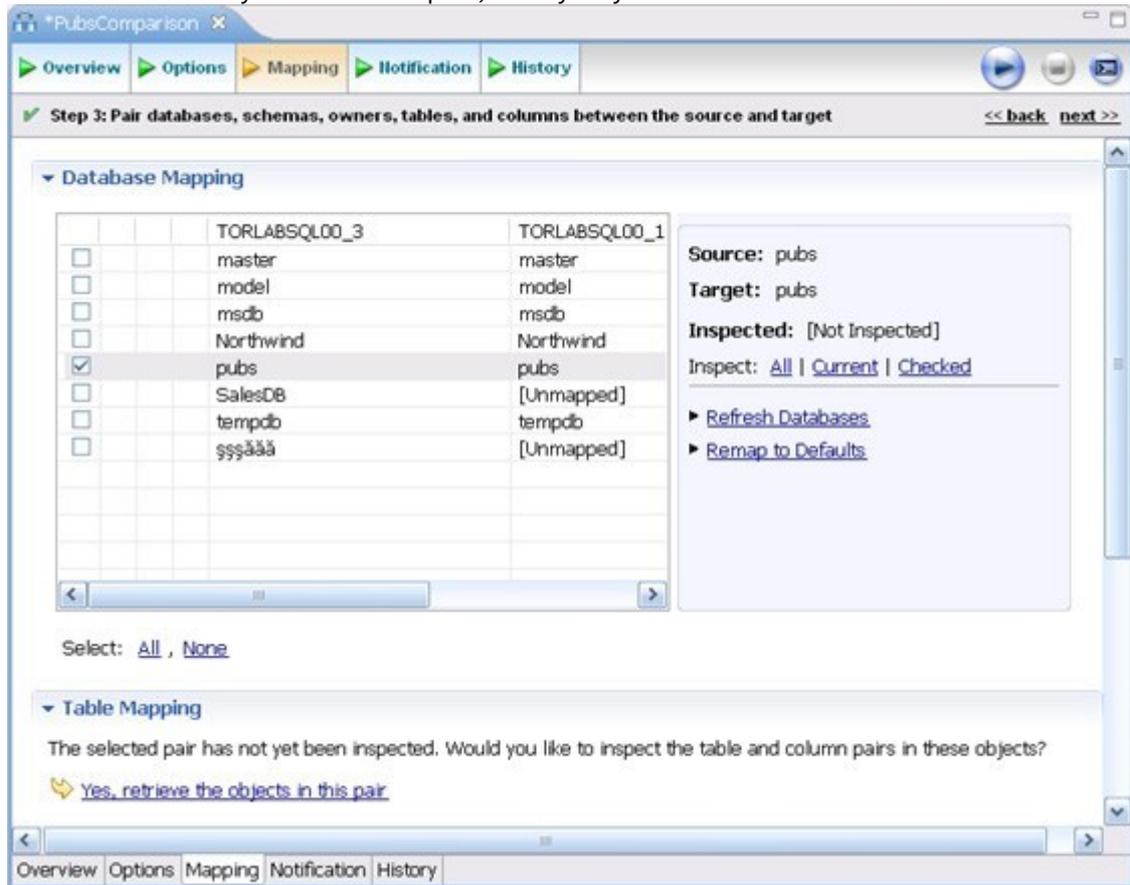
2. In the Options tab, review the various settings available to you. In particular, select or de- select the check boxes in the Mapping Options section.



To refine comparisons in the Mapping tab

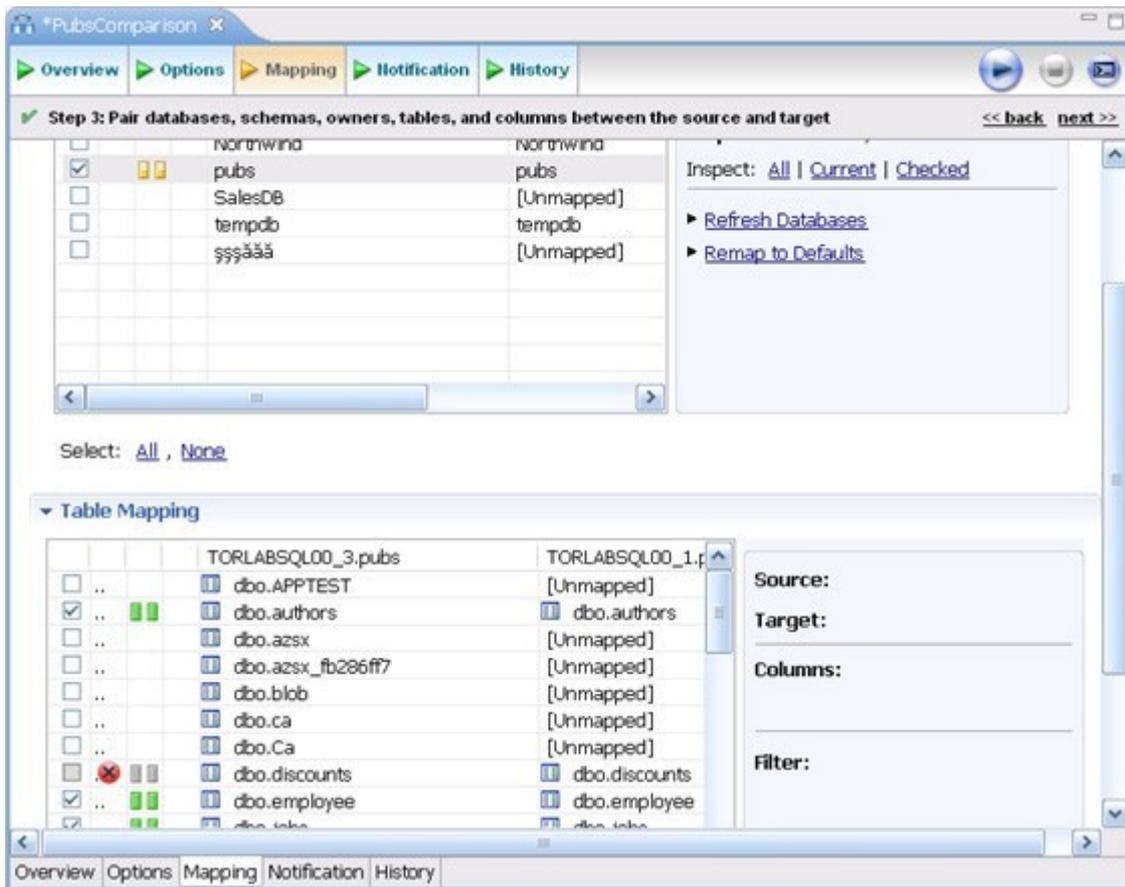
- Click the Mapping tab.

- Select the databases you want to compare; usually they have the same or a similar name.

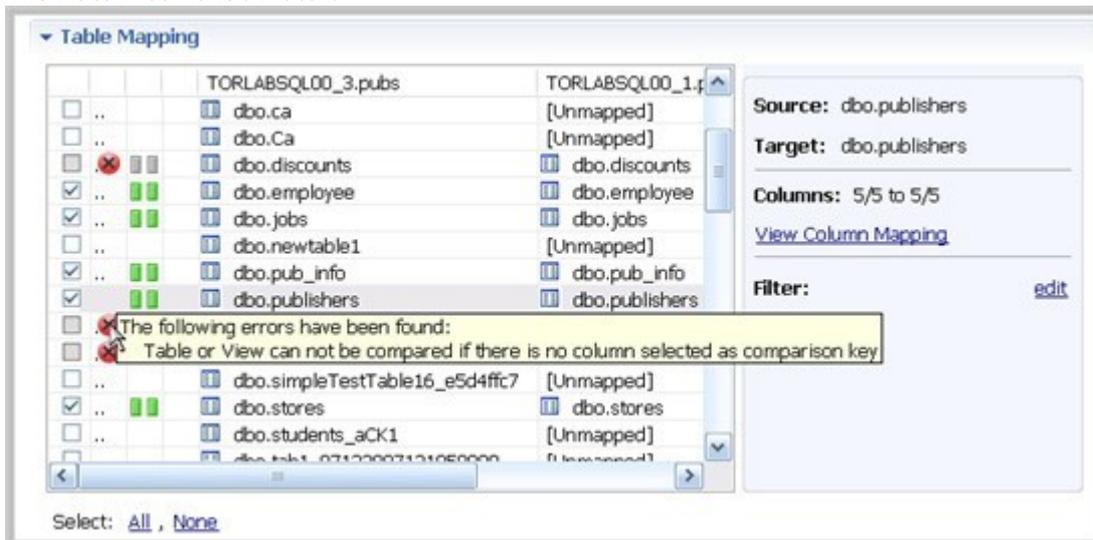


i Use the **All** and **None** commands at the bottom of the pane to select all or none of the databases.

- By default, DB Change Manager does not show the table mappings. To see them, select the **Inspect: Checked** option on the right side of the **Database Mapping** pane. The progress bar displays in the lower right corner. The **Table Mapping** pane shows how and whether the tables map to each other in the two databases.



The **Data Comparison Job Editor** automatically maps and selects databases and tables between the source and target when it detects correspondences. It indicates when there is no mapping and displays information when it cannot make a match.

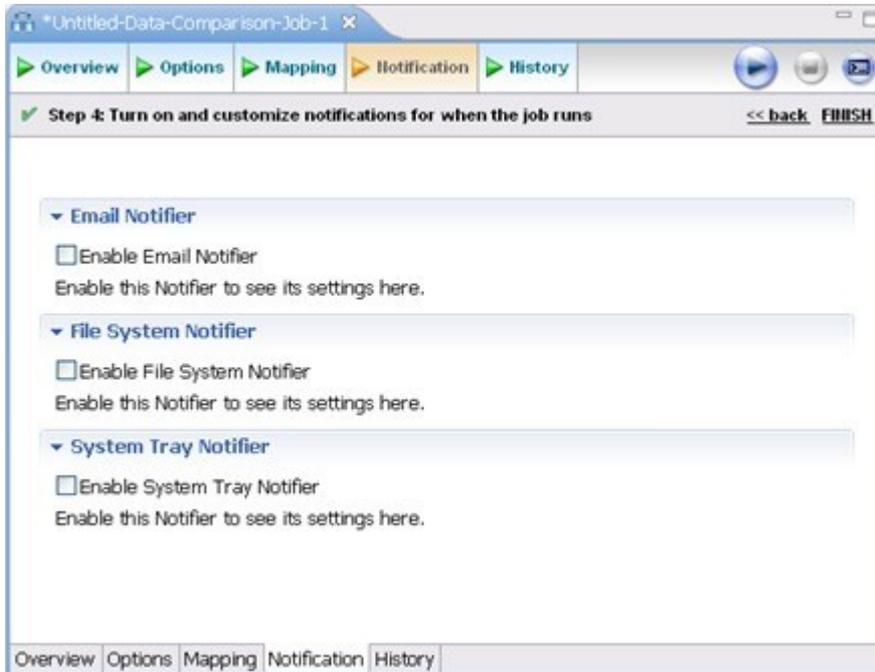


- Select or de-select the check box next to each mapping, as needed. You can adjust incomplete mappings before running the job. Click undefined items in the Target data source column and use the drop down menu to map them.

Click **Filter: edit** while a pair is selected to use the Columns, Comparison Key, and Filter dialog and further refine the mappings, such as inserting a WHERE clause.

To specify notifiers in the Notification tab

1. Click the Notification tab.



2. In the Notification tab, select where DB Change Manager sends notifications when this job is run.

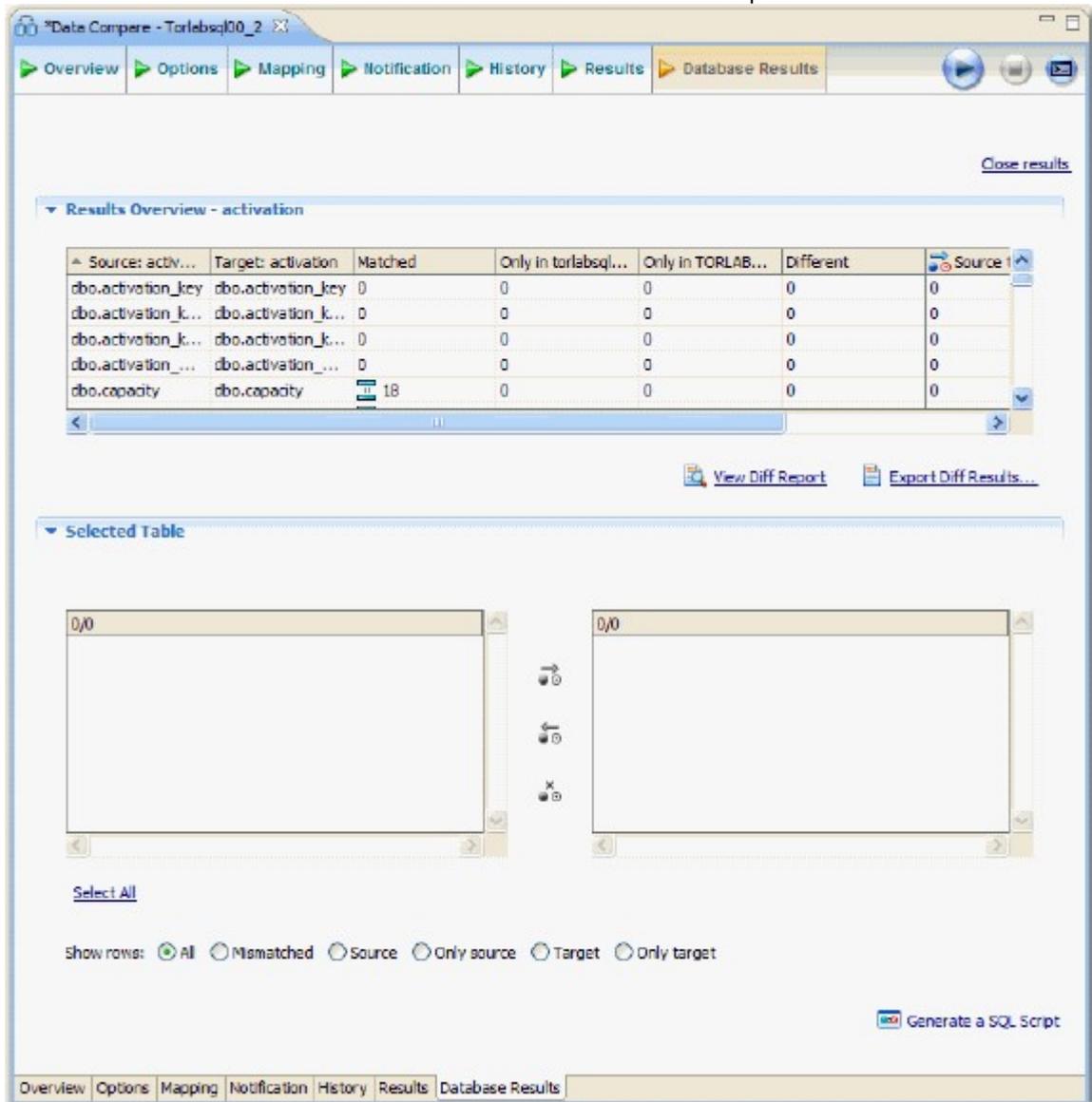
To run the job and see the results

1. Click the **Run Job** icon located in the upper-right side of the window.



The Results tab displays the job progress bar and the results, when available.

2. Click **View** in the **Results** column to view information for an individual pair.



3. In the Database Results tab, review the detailed results for the tables and rows of the selected databases.

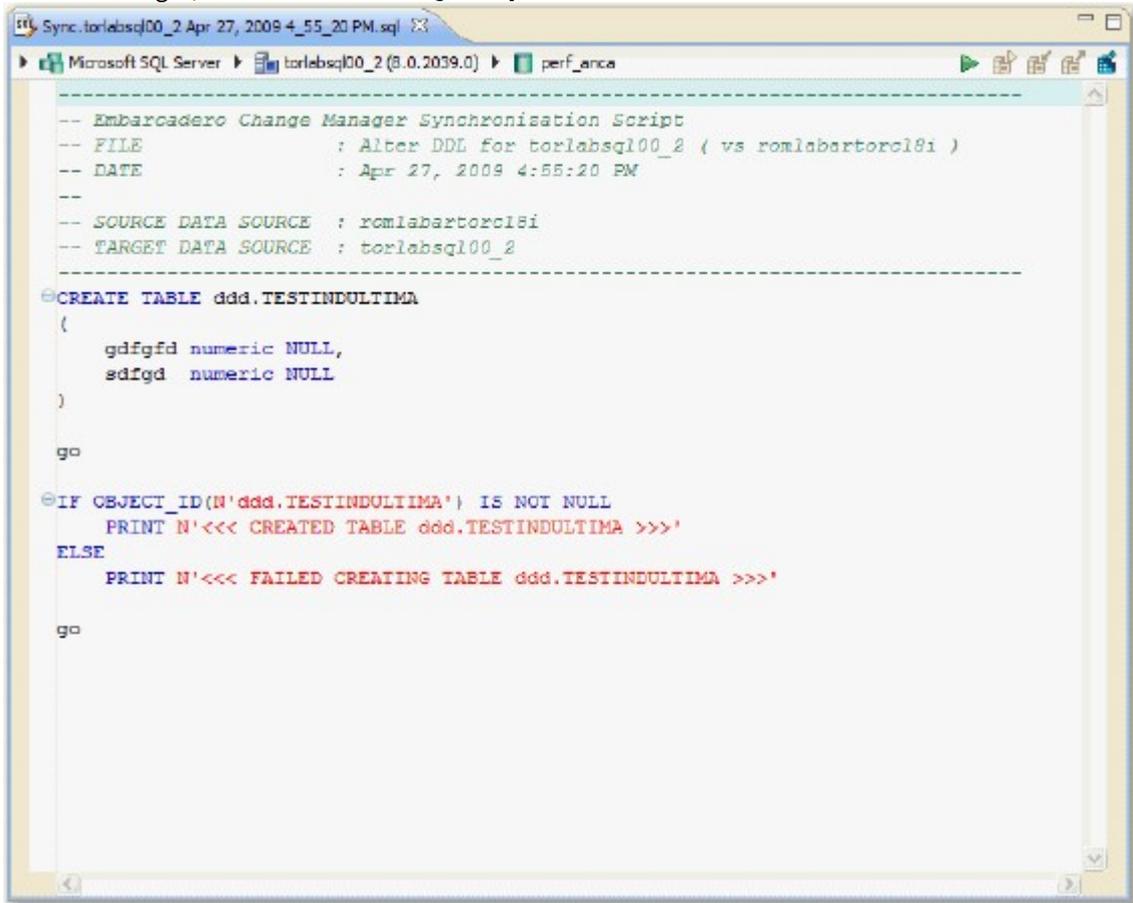
Tutorial 2: Synchronizing data

⚠ Before working with tutorials, identify two data sources that you can use for practice. These data sources should not be live or mission-critical, and they should be similar in schema and data.

To generate and run an SQL synchronization script

1. At the bottom of the Database Results tab of a database comparison job, click one of the radio buttons, such as Mismatched, to specify the type of resolution to perform.

- At the lower right, click **Generate an SQL Script**.



- In the SQL Editor, click the Execute SQL icon, the green arrow, to synchronize the data.

i Executing an SQL script may change the data in your data sources. Be sure to review all potential changes, and read the script, before running the script.

- If you want to create a report, select the Database Results tab, and then click **View Diff Report**.

Highlighted Feature: The Compression Comparison Option

Comparing large repositories with many databases, tables, and rows can be a time consuming and tedious process, especially when you require row-level accuracy.

The Compression Comparison option enables an algorithm that causes the data comparison to initially execute at a table level instead of the row level. Each table in a pair is assigned a value and if the values match the comparison process moves to the next pair of tables. If the values do not match, the process performs a more detailed comparison at the row level. This allows for a faster comparison process in general, without losing any accuracy during the process.



To enable this feature, navigate to the **Options tab > Execution Options** section, and then select **Compression Comparison**.

Tutorial 3: Comparing configurations

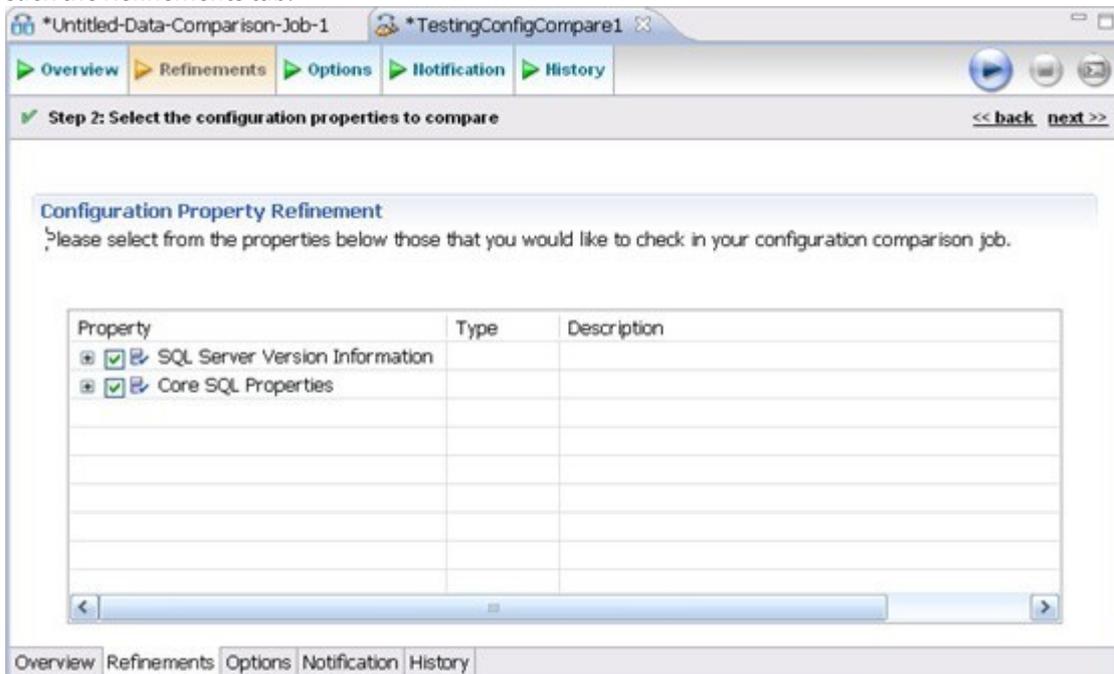
⚠ Before working with tutorials, identify two data sources that you can use for practice. These data sources should not be live or mission-critical, and they should be similar in schema and data.

Configuration comparison jobs examine the configuration settings of two specified data sources and describe the differences between them. You select a pair of data sources from Data Source Explorer, refine the property comparisons, run the job, and view the results. Based on the results you can correct any issues between the pair.

You can also compare a data source against a configuration archive or standard. Archives are a snapshot of the configuration settings of a data source at a point in time. Standards are customized sets of properties with threshold operators that help you create optimal configurations; against which you can compare existing data source settings.

To define and run a configuration comparison job

1. Select **File > New > Configuration Comparison Job**. If prompted to select a project, select the project you created earlier.
2. Type the name of the job in the **Name** field.
3. Drag and drop a data source from Data Source Explorer to the **Configuration Comparison Source** box.
4. Drag and drop a data source from Data Source Explorer to the **Configuration Comparison Target** box.
5. Click the Refinements tab.

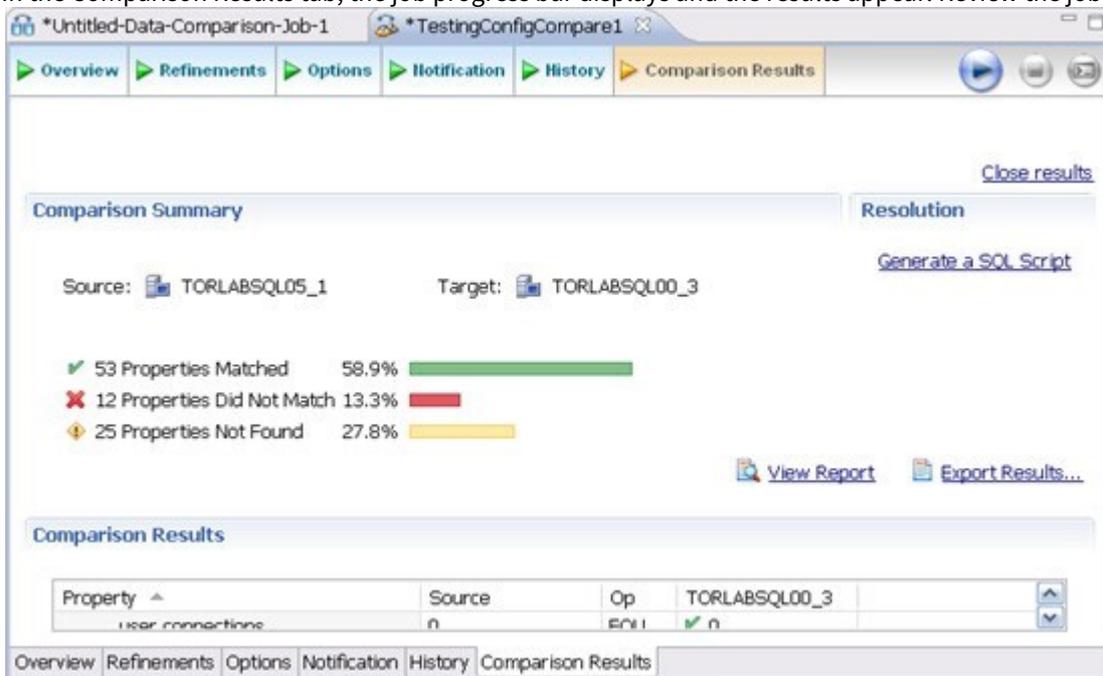


6. Keep the defaults in the Refinements tab.
7. Click the Options tab.

- In the Options tab, select the **Extract DDL** check box.



- Click the **Run Job** icon.
- In the Comparison Results tab, the job progress bar displays and the results appear. Review the job results.



In the Comparison Results tab, you can generate an SQL script to synchronize the configurations just as in [Tutorial 2: Synchronizing data](#).

Highlighted feature: Comparing multiple targets

You can compare multiple target sources against a single source. In the Configuration Comparison Target box of the Overview tab, click the **Add More Targets** option. The box changes to a column list view, where you can drag and drop additional sources from Data Source Explorer to the Configuration Comparison Target box.

Tutorial 4: Defining a configuration standard

- Before working with tutorials, identify two data sources that you can use for practice. These data sources should not be live or mission-critical, and they should be similar in schema and data.

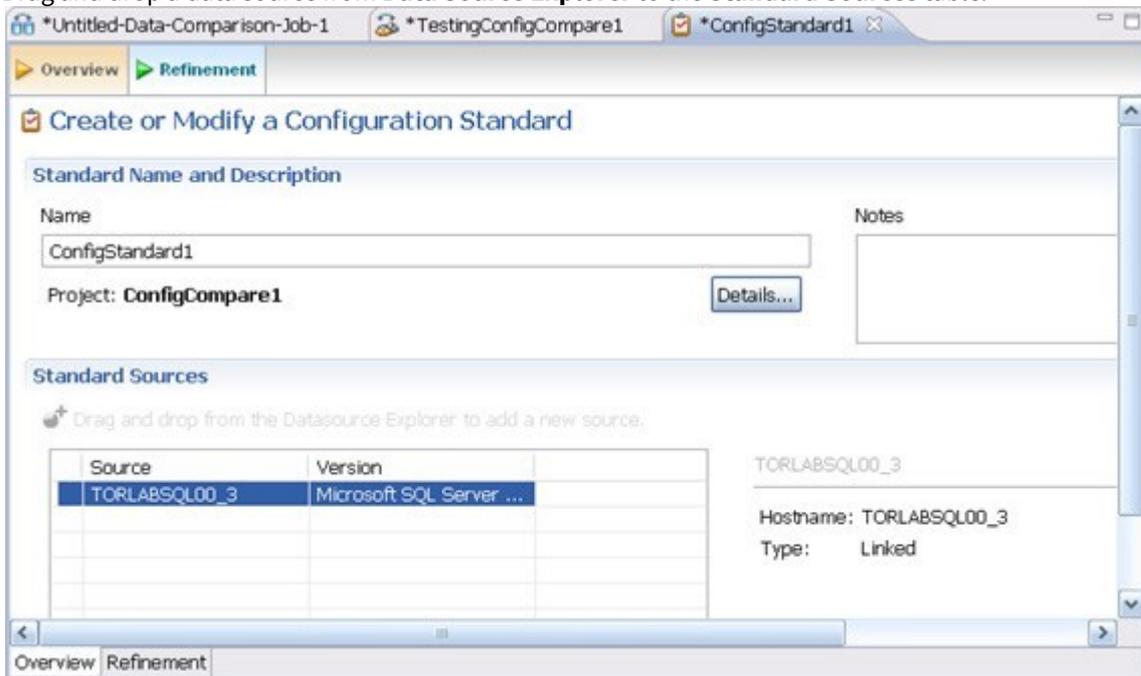
A configuration standard is composed of properties and values that you define:

- Configuration properties (static or dynamically updated based on the source values)
- Values that include threshold comparison operators (greater than, less than, falls within or is a member of a specified set, etc.)

You can use a standard in configuration comparison jobs against data sources and archives in order to determine if the target conforms to the standard. A standard can only be used as a source in comparison jobs.

To define a configuration standard

1. Select **File > New > Configuration Standard**.
2. If prompted, select a project to use.
3. In the Overview tab, enter the name of the standard in the **Name** field.
4. Drag and drop a data source from **Data Source Explorer** to the **Standard Sources** table.



5. In the Refinement tab, define each listed property using the **Source**, **Operator**, and **Value** controls.
6. Select **File > Save** to save the standard. It appears in the Project Explorer.

You can subsequently drag and drop the new standard from **Project Explorer** to the **Comparison Source** box in the [Configuration Comparison Job Editor](#). You can then run a job using the standard as you would use a registered data source.

Tutorial 5: Comparing and synchronizing schemas

⚠ Before working with tutorials, identify two data sources that you can use for practice. These data sources should not be live or mission-critical, and they should be similar in schema and data.

Once you create a baseline archive of your schema, you can compare it against the other schemas, even in live environments. Also, you can synchronize the schema of a data source with the archive using the [Schema Comparison Job editor](#).

To define a schema comparison job

1. Select **File > New > Schema Comparison Job** and if prompted, select a project.
2. In the Schema Comparison Job editor, type the name of the job in the **Name** field.
3. Drag and drop a data source from **Data Source Explorer** to the **Schema Comparison Source** box.
4. Drag and drop a data source from **Data Source Explorer** to the **Schema Comparison Target** box.
5. Click the Mapping tab.
6. In the Mapping tab, review the object mappings that are already set up for you. Change the target mappings as needed.
7. In the Options tab, select or de-select the configuration properties.
8. Click the **Run Job** icon.
9. In the Comparison Results tab, the job progress bar displays and the results appear. Review the job results.

To generate and run a schema synchronization script

1. In the Comparison Results tab, click **Generate Sync Script** in the Resolution column of the Comparison Results table. DB Change Manager generates a synchronization script in the SQL Editor.
2. Modify the script as needed, and save it using the **File > Save** command.
3. In the SQL Editor, select the Execute SQL icon to synchronize the data.

 Executing the SQL script changes the schema in your target data source. Be sure to review all potential changes and read the script before running it.

4. If you want to create a report, click **Generate Report**. The Export dialog opens and provides additional report options.

Tutorial 6: Defining a schema archive

 Before working with tutorials, identify two data sources that you can use for practice. These data sources should not be live or mission-critical, and they should be similar in schema and data.

A schema archive contains a snapshot of the schema of a registered data source. You compare schema archives against other data sources and archives to identify possible inconsistencies.

A schema archive is generated from a single data source at a specific point in time. Like a configuration archive, a schema archive has versioning functionality and can be periodically updated with the current values of the data source. Older versions are retained and can be recalled as needed.

You create a schema archive using the [Schema Archive Job editor](#). Once created and saved, it is listed in the Data Source Explorer.

To define a schema archive

1. Right-click on a data source in Data Source Explorer and select **Archive > Schema Archive Job**. Select a project, if prompted.
2. In the Archive Job editor, enter the name of the archive in the **Name** field.
3. In the Refinements tab, narrow the job down by selecting owners, object types, and specific objects to include.
4. Click the **Run Job** icon.
5. In the New Unsaved Version tab, the results appear. Review the job results.
6. Click **File > Save**. The Unsaved Version tab changes to Version 1, and Version 1 appears in the Datasource Explorer tab under the datasource you chose.

 You can create more versions by clicking the **Run Job** icon again.

You can subsequently drag and drop the new archive version from Data Source Explorer to the **Source** or **Target** boxes in the [Schema Comparison Job editor](#). You can now run a job using the archive just as you would use a data source.

Tutorial 7: Masking data

⚠ Before working with tutorials, identify two data sources that you can use for practice. These data sources should not be live or mission-critical, and they should be similar in schema and data.

Data masking is a way of securing sensitive data during the development or testing phases of a database development project. It is often performed as a security or compliance measure that protects important information. By masking valid production data, you can provide a copy of the data that is “scrambled” but still represents your production environment.

To mask data

1. Select **File > New > Data Comparison Job** and if prompted, select a project.
2. Provide a name of the job in the **Name** field and then drag and drop datasources from the **Data Source Explorer** to the **Data Comparison Source** and **Data Comparison Target** boxes.
3. In the Options tab, ensure that **Automatically Synchronize** is selected and read the description shown for that option.
4. In the Mapping tab, select a row corresponding to a matching database pair in the **Database Mapping** box, and then select **Yes retrieve the objects in this pair**.

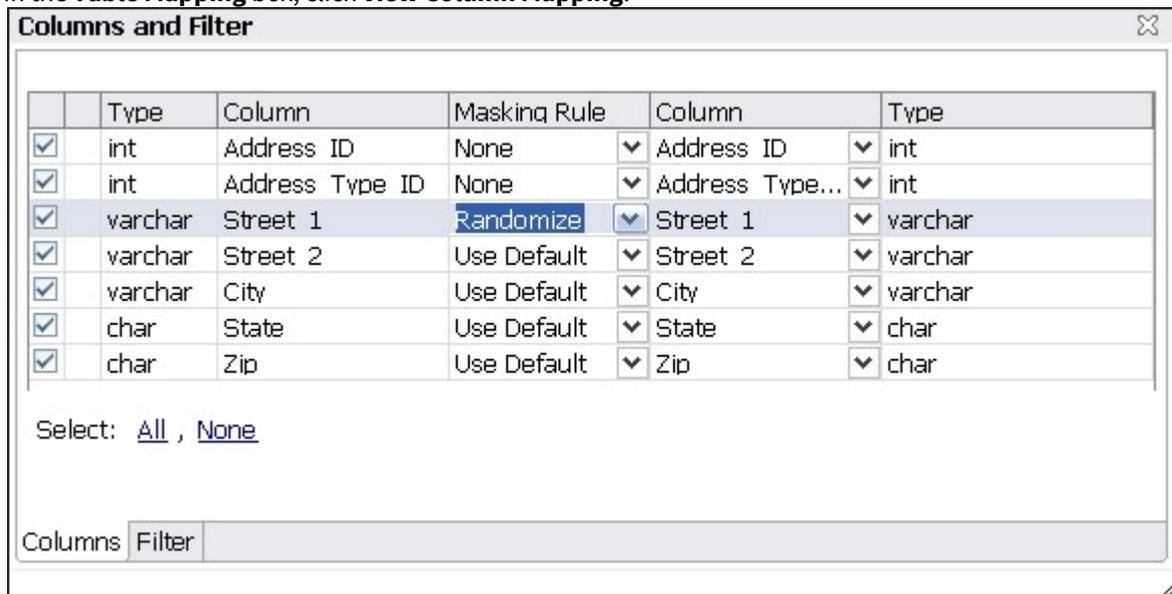
The screenshot displays the 'Table Mapping' window. On the left, a table lists database objects from two sources: TORLABSQL05_1.FranklynA and TORLABSQL05_2.F. The objects are mapped one-to-one. The right-hand panel shows configuration for the selected mapping (Source: dbo.Address, Target: dbo.Address). The 'Default Masking Rule' is set to 'Use Default'. Other options include 'Apply to All', 'Columns: No Masking', and a 'Filter' field.

	TORLABSQL05_1.FranklynA	TORLABSQL05_2.F
<input checked="" type="checkbox"/>	dbo.Address	dbo.Address
<input checked="" type="checkbox"/>	dbo.Category	dbo.Category
<input checked="" type="checkbox"/>	dbo.Customer	dbo.Customer
<input checked="" type="checkbox"/>	dbo.Customer_Address	dbo.Customer_
<input checked="" type="checkbox"/>	dbo.Customer_Order	dbo.Customer_C
<input checked="" type="checkbox"/>	dbo.Customer_Order_Product	dbo.Customer_C
<input checked="" type="checkbox"/>	dbo.Product	dbo.Product
<input checked="" type="checkbox"/>	dbo.Ref_Address_Type	dbo.Ref_Addres
<input checked="" type="checkbox"/>	dbo.Supplier	dbo.Supplier

Source: dbo.Address
 Target: dbo.Address
 Columns: 7/7 to 7/7
[View Column Mapping](#)
 Default Masking Rule:
 Use Default
[Apply to All](#)
 Columns: No Masking
 Filter: [edit](#)
 [no filter]

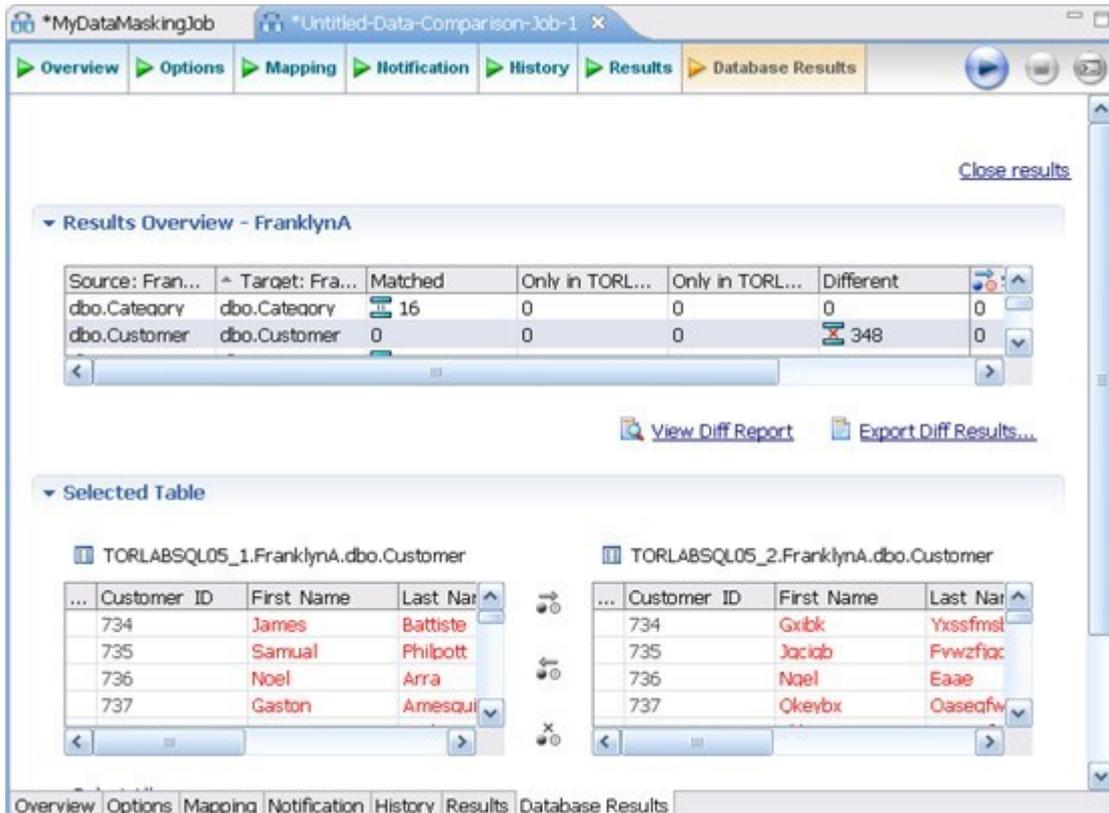
Select: [All](#), [None](#)

5. In the **Table Mapping** box, click **View Column Mapping**.



6. Click on the **Data Masking Rule** option for a column and select one of the following:
- **Use Default.** The default masking rule as set in the Database Mapping box.
 - **None.** No masking is applied.
 - **Randomize.** Replaces characters with other random characters.
 - **Shuffle.** Reorders the values in different rows; for example, in a **First Name** column, Bob, Ted, and Alice can change to Alice, Bob, and Ted.
7. Close the window.
8. Continue to experiment with selection and settings until you are comfortable with masking options.
9. Click **File > Save** to save the job.

10. Click the **Run Job** icon.



The Database Results tab shows the masked data.

Tutorial 8: Defining and running automated scripts

⚠ Before working with tutorials, identify two data sources that you can use for practice. These data sources should not be live or mission-critical, and they should be similar in schema and data.

When you need to repeatedly run the same job, you can set up automated tasks that execute the jobs on a regular schedule. DB Change Manager helps create a command line executable for any job. You can run the command line executable on an ad hoc basis. You can also group the commands into a batch file and run them.

To define command line syntax and generate batch files

1. Open the job for which you want to define an execution script.
2. Select the **Command Line Syntax** icon in the upper right of any tab.
3. In the Script Generation Wizard, add output report options.
4. Specify the **Directory** where the files are generated.
5. Click **Next**.
6. Specify the **Location** where the batch file is generated.
7. Specify the file format for the script.
8. Click **Finish**.

DB Change Manager generates the file and saves it to the location you specified.

If you want to add multiple jobs to the batch file, generate a batch file as normal, and then copy the script syntax to the clipboard for each job you want to add to the batch file. You can then add them to the existing batch file manually.

